

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
COORDENADORIA DO CURSO DE ENGENHARIA DE SOFTWARE

DIOGO REIS PAVAN

**APLICAÇÃO DE UM *FRAMEWORK* CONCEITUAL DE GESTÃO DO
CONHECIMENTO PARA MELHORIA EM DEVOPS**

MONOGRAFIA

DOIS VIZINHOS

2018

DIOGO REIS PAVAN

**APLICAÇÃO DE UM *FRAMEWORK* CONCEITUAL DE GESTÃO DO
CONHECIMENTO PARA MELHORIA EM DEVOPS**

Monografia apresentada ao Coordenadoria do Curso de Engenharia de Software da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de Bacharel em Engenharia de Software.

Orientador: Prof. Dr. Rafael Alves Paes De Oliveira

Co-orientador: Prof^ª. Dr^ª. Érica Ferreira De Souza

DOIS VIZINHOS

2018



TERMO DE APROVAÇÃO

Aplicação de um framework conceitual de gestão do conhecimento para melhoria em DevOps

por

Diogo Reis Pavan

Este Trabalho de Conclusão de Curso foi apresentado em 03 de Dezembro de 2018 como requisito parcial para a obtenção do título de Bacharel em Engenharia de Software. O(a) candidato(a) foi arguido(a) pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Rafael Alves Paes de Oliveira
Presidente da Banca

Alinne Cristinne Correa Souza
Membro Titular

Pedro Henrique de Alencar Machado
Membro Titular

* A Folha de Aprovação assinada encontra-se na Coordenação do Curso

Dedico este trabalho a todos que colaboraram com o desenvolvimento desse estudo.

AGRADECIMENTOS

Agradeço aos meus pais, irmão, namorada e amigos que me apoiaram e fizeram parte dessa jornada.

Agradeço aos professores, em especial meus orientadores Rafael Alves Paes de Oliveira e Érica Ferreira de Souza, que colaboraram para o desenvolvimento desse trabalho.

Por fim, agradeço a Deus por mais essa etapa vencida.

RESUMO

PAVAN, Diogo R.. APLICAÇÃO DE UM *FRAMEWORK* CONCEITUAL DE GESTÃO DO CONHECIMENTO PARA MELHORIA EM DEVOPS. 101 f. Monografia – Coordenadoria do Curso de Engenharia de Software, Universidade Tecnológica Federal do Paraná. Dois Vizinhos, 2018.

O DevOps (Desenvolvimento e Operações) é um movimento cultural que visa aproximar a equipe de desenvolvimento, responsável pelas novas funcionalidades e produtos, e a equipe de operações, incumbida pela implantação e manutenção do sistema. Tal cultura é apoiada por ferramentas e práticas que possibilitam a automação e a agilidade no desenvolvimento e entrega de software com qualidade. Entretanto, a adoção efetiva da cultura DevOps é afetada por: (i) ausência de estratégias advindas do meio acadêmico para o profissional; (ii) ausência da gestão do conhecimento nas organizações; e (iii) falta de consciência das organizações que o DevOps é um movimento cultural. Isto posto, este estudo possui o objetivo principal de fornecer um insumo que apoie à adoção da cultura DevOps em cenários práticos de desenvolvimento de software, por meio do uso de recursos de Gestão do Conhecimento. Adicionalmente, é apresentada uma investigação acerca do cenário da cultura DevOps e da Gestão do Conhecimento em um âmbito nacional. Para tanto, foram executadas duas avaliações empíricas: (i) um estudo de caso em cenário prático de desenvolvimento de software, utilizando o *Framework* DOKS para a avaliação e apoio das práticas e princípios do DevOps com foco na gestão do conhecimento; e (ii) aplicação de uma *Survey* em nível nacional para mapear práticas, benefícios e limitações ao adotar a cultura DevOps. Os resultados obtidos com as atividades empíricas foram: (i) o estudo de caso realizado auxiliou na melhoria do cenário DevOps e da Gestão de Conhecimento no objeto participante do estudo; e (ii) a *Survey* possibilitou uma avaliação parcial e inicial do cenário DevOps e da Gestão do Conhecimento em um nível nacional.

Palavras-chave: Processo de Software, DevOps, Desenvolvimento e Operações, Gestão do Conhecimento, Engenharia de Software

ABSTRACT

PAVAN, Diogo R.. APPLYING A CONCEPTUAL KNOWLEDGE MANAGEMENT FRAMEWORK TO DEVOPS IMPROVEMENT. 101 f. Monografia – Coordenadoria do Curso de Engenharia de Software, Universidade Tecnológica Federal do Paraná. Dois Vizinhos, 2018.

DevOps (Development and Operations) is a cultural movement that aims to approximate the development team, which is responsible for new functionalities and products, and the operation team, which is responsible to implant and maintain the systems. Such culture is supported by tools and practices that make it possible the automation and agility on developing and delivering high-quality software. However, the effective adoption of DevOps is affected by: (i) lack of academic efforts to support the industrial environments; (ii) lack of knowledge management in software organizations; (iii) lack of awareness of organizations that DevOps is a cultural movement. Thus, the main goal of this study is to provide information to support the DevOps adoption in real-world scenarios of software development, by means of Knowledge Management. In addition to that, it is presented an investigation about the DevOps cultural scenario and knowledge management in the national context. Then, two empirical evaluation were conducted: (i) a case study in a practical software development scenario, exploring the DOKS framework to the evaluation and support of DevOps practices and principles with focus on the knowledge management; and (ii) applying a survey in the national context in order to identify practices, benefits, and limitations on adopting the DevOps culture. Our results have revealed: (i) the study case helped in the improvement of the DevOps scenario and the Knowledge Management in the study object; and (ii) the survey made it possible for a partial-initial evaluation of DevOps and Knowledge Management at a national level.

Keywords: Software Process, DevOps, Development and Operations, Knowledge Management, Software Engineering

LISTA DE FIGURAS

FIGURA 1	– Representação do Modelo em Cascata	20
FIGURA 2	– Representação do Modelo em V	21
FIGURA 3	– Representação do Modelo Incremental	22
FIGURA 4	– Representação do fluxo da Prototipagem	22
FIGURA 5	– Representação do Modelo Espiral	23
FIGURA 6	– Representação do <i>Framework</i> Scrum	25
FIGURA 7	– Representação do XP	26
FIGURA 8	– Pilares do DevOps	29
FIGURA 9	– Representação do <i>framework</i> CAMS	31
FIGURA 10	– Processo SECI	33
FIGURA 11	– DOKS <i>Framework</i>	36
FIGURA 12	– Representação da legenda e da avaliação do <i>Framework</i> DOKS	36
FIGURA 13	– Fluxo das atividades do Estudo de Caso	40
FIGURA 14	– Gráficos sobre a experiência dos funcionários na área de TI e na Empresa A	47
FIGURA 15	– Gráfico sobre a questão de ter ouvido falar do termo DevOps	48
FIGURA 16	– Gráficos sobre a experiência dos funcionários com o DevOps	49
FIGURA 17	– Gráfico sobre as práticas, ferramentas e tecnologias do DevOps na Empresa	50
FIGURA 18	– Gráfico sobre a existência da GC nas equipes	51
FIGURA 19	– Gráficos sobre as ferramentas e práticas da GC na Empresa A	51
FIGURA 20	– Gráfico das respostas sobre ações de Melhoria contínua e reflexões	53
FIGURA 21	– Gráfico das respostas sobre ações para a cultura DevOps	54
FIGURA 22	– Gráficos sobre as ações para GC e blog	54
FIGURA 23	– Gráfico das respostas sobre monitoramento e otimização do sistema	55
FIGURA 24	– Avaliação dos Princípios do <i>Framework</i> DOKS	57
FIGURA 25	– Avaliação das práticas e ferramentas do Dev no <i>Framework</i> DOKS	58
FIGURA 26	– Avaliação das práticas e ferramentas do Ops no <i>Framework</i> DOKS	58
FIGURA 27	– Avaliação da GC no <i>Framework</i> DOKS	59
FIGURA 28	– Gráficos sobre o perfil do profissional	59
FIGURA 29	– Gráficos sobre o perfil da empresa	60
FIGURA 30	– Gráficos sobre a experiência e conhecimento da cultura DevOps do profissional	61
FIGURA 31	– Gráfico sobre a experiência dos colegas com o DevOps	61
FIGURA 32	– Gráficos acerca da Cultura DevOps na empresa	62
FIGURA 33	– Gráficos acerca da Implantação Contínua	63
FIGURA 34	– Gráficos sobre o Controle de Versão da Empresa	64
FIGURA 35	– Gráficos sobre os Testes Automatizados e Contínuos	64
FIGURA 36	– Gráficos sobre a Integração Contínua	65
FIGURA 37	– Gráficos sobre Serviços na Nuvem	65
FIGURA 38	– Gráfico sobre as práticas, ferramentas ou tecnologias adicionais que dão suporte para o DevOps	66
FIGURA 39	– Gráficos sobre aspectos gerais da GC na empresa	68

FIGURA 40	– Gráfico sobre os benefícios da adoção da Cultura DevOps	70
FIGURA 41	– Gráfico sobre as limitações na adoção da Cultura DevOps	72
FIGURA 42	– Visão macro do <i>framework</i> para avaliação e melhoria do processo de software	78
FIGURA 43	– Representação das fases do <i>framework</i> de avaliação integrada	79
FIGURA 44	– <i>Framework</i> para times ágeis	80
FIGURA 45	– Representação do <i>Framework</i> para compartilhamento de conhecimento em organizações Globais de Desenvolvimento de Software	81

LISTA DE QUADROS

QUADRO 1 –	Principais Ferramentas e técnicas para GC	35
QUADRO 2 –	Relação entre atividades empíricas e objetivos específicos	38
QUADRO 3 –	Relacionamento entre os tópicos e questões de pesquisa da <i>Survey</i>	44
QUADRO 4 –	Relação entre as lacunas e as sugestões de melhorias	46
QUADRO 5 –	Definição do termo DevOps pelo funcionário	48
QUADRO 6 –	Legenda da pergunta sobre práticas, ferramentas ou tecnologias adicionais do DevOps	50
QUADRO 7 –	Comentário adicional sobre a GC	51
QUADRO 8 –	Comentários adicionais sobre a Melhoria contínua e reflexões	53
QUADRO 9 –	Comentários adicionais sobre o blog e GC	55
QUADRO 10 –	Comentários adicionais sobre monitoramento e otimização do sistema ..	56
QUADRO 11 –	Comentários adicionais sobre a pesquisa	56
QUADRO 12 –	Legenda da pergunta sobre a caracterização da cultura de trabalho da empresa	62
QUADRO 13 –	Legenda da pergunta sobre práticas, ferramentas ou tecnologias adicionais do DevOps	66
QUADRO 14 –	Respostas sobre as práticas, ferramentas ou tecnologias adicionais que dão suporte para o DevOps	67
QUADRO 15 –	Respostas sobre as principais ações realizadas pela empresa voltadas para a GC	69
QUADRO 16 –	Legenda da pergunta sobre benefícios da cultura DevOps	70
QUADRO 17 –	Respostas sobre os benefícios e comentários adicionais da adoção da Cultura DevOps	71
QUADRO 18 –	Legenda da pergunta sobre limitações na adoção da cultura DevOps	72
QUADRO 19 –	Respostas sobre as limitações e comentários adicionais na adoção da Cultura DevOps	72
QUADRO 20 –	Relação entre práticas DevOps e ferramentas para executá-las	75

LISTA DE SIGLAS

AWS	<i>Amazon Web Services</i>
CAMS	<i>Culture, Automation, Measurement, Sharing</i> - Cultura, Automação, Medição, Compartilhamento
CMMI	Capability Maturity Model Integration - Modelo Integrado de Maturidade em Capacitação
DevOps	<i>Development and Operations</i> - Desenvolvimento e Operações
DOKS	<i>DevOps Knowledge Sharing</i>
ERES	Escola Regional de Engenharia de Software
ES	Engenharia de Software
GC	Gestão do Conhecimento
IEEE	<i>Institute of Electrical and Electronics Engineers</i> - Instituto de Engenheiros Eletricistas e Eletrônicos
ISO/IEC	<i>International Organization of Standardization and International Electrotechnical Commission</i> - Organização Internacional para Padronização e Comissão Eletrotécnica Internacional
MA-MPS	Método de Avaliação para Melhoria de Processo de Software
MR-MPS	Modelo de Referência para Melhoria de Processo de Software
SBC	Sociedade Brasileira de Computação
SCAMPI	<i>Standard CMMI Appraisal Method for Process Improvement</i> - Método de Avaliação Padrão do CMMI para Melhoria de Processos
SEBRAE	Serviço Brasileiro de Apoio às Micro e Pequenas Empresas
SWEBOK	<i>Software Engineering Body of Knowledge</i>
UTFPR	Universidade Tecnológica Federal do Paraná
XP	<i>Extreme Programming</i>

SUMÁRIO

1	INTRODUÇÃO	13
1.1	CONTEXTO	13
1.2	PROBLEMA DE PESQUISA	14
1.3	OBJETIVOS	15
1.3.1	Objetivo Geral	15
1.3.2	Objetivos Específicos	16
1.4	JUSTIFICATIVA	16
1.5	ORGANIZAÇÃO DA MONOGRAFIA	17
2	FUNDAMENTAÇÃO TEÓRICA	18
2.1	ENGENHARIA DE SOFTWARE	18
2.2	PROCESSOS DE DESENVOLVIMENTO DE SOFTWARE	19
2.2.1	Processos e Metodologias de Desenvolvimento Tradicionais	19
2.2.2	Metodologias Ágeis	23
2.3	DEVOPS	26
2.3.1	História do DevOps	27
2.3.2	Aspectos conceituais de DevOps	28
2.3.2.1	Pilares do DevOps	28
2.3.2.2	Cultura, Automação, Medição e Compartilhamento	30
2.3.3	Práticas de DevOps	30
2.4	GESTÃO DO CONHECIMENTO	32
2.4.1	Técnicas e Ferramentas de Gestão de Conhecimento	34
2.5	DOKS <i>FRAMEWORK</i>	35
3	AVALIAÇÕES EMPÍRICAS	38
3.1	PLANEJAMENTO EMPÍRICO	38
3.2	ESTUDO DE CASO	39
3.2.1	<i>Design</i> da Pesquisa	39
3.2.2	Execução do Estudo de Caso	40
3.2.3	Estratégia para Análise dos Dados	41
3.3	PLANEJAMENTO E EXECUÇÃO DA <i>SURVEY</i>	42
3.3.1	Questões de Pesquisa	43
3.3.2	Estruturação da <i>Survey</i>	43
3.3.3	Estratégias de Coleta e Análise dos Dados	44
4	RESULTADOS E DISCUSSÕES	45
4.1	RESULTADOS DO ESTUDO DE CASO	45
4.1.1	Lacunas identificadas e Melhorias propostas	45
4.1.2	Questionário 1	47
4.1.3	Questionário 2	52
4.1.4	<i>Framework</i> DOKS	56
4.1.4.1	Princípios	56
4.1.4.2	Práticas e Ferramentas	57
4.1.4.3	Gestão de Conhecimento	58

4.2	RESULTADOS DA <i>SURVEY</i>	59
4.2.1	Perfil do profissional e da empresa	59
4.2.2	Experiência com DevOps	60
4.2.3	Cultura DevOps na empresa	61
4.2.4	Práticas e Ferramentas DevOps na empresa	63
4.2.5	Gestão de conhecimento na empresa	67
4.2.6	Benefícios na adoção da cultura DevOps	70
4.2.7	Limitações na adoção da cultura DevOps	71
4.3	RESPOSTAS ÀS QUESTÕES DE PESQUISA E DISCUSSÕES	73
4.3.1	Discussões sobre o Estudo de Caso	73
4.3.1.1	Questão de Pesquisa do Estudo de Caso	73
4.3.2	Discussões sobre a <i>Survey</i>	74
4.3.2.1	Questão de Pesquisa 1	74
4.3.2.2	Questão de Pesquisa 2	74
4.3.2.3	Questão de Pesquisa 3	75
4.3.2.4	Questão de Pesquisa 4	75
4.3.2.5	Questão de Pesquisa 5	75
4.3.2.6	Questão de Pesquisa 6	75
4.4	AMEAÇAS À VALIDADE	76
4.5	PUBLICAÇÕES	77
5	TRABALHOS RELACIONADOS	78
5.1	<i>FRAMEWORKS</i> PARA PROCESSOS	78
5.2	<i>FRAMEWORKS</i> PARA A GESTÃO DO CONHECIMENTO	79
6	CONCLUSÃO	82
	REFERÊNCIAS	84
	Apêndice A – PROTOCOLO DE ESTUDO DE CASO	88
	Apêndice B – <i>SURVEY</i> NA ÍNTEGRA	97
	Anexo A – TABELA PERÍODICA DE FERRAMENTAS DEVOPS	100

1 INTRODUÇÃO

1.1 CONTEXTO

A indústria de software busca constantemente maneiras de acelerar a entrega de produtos de alta qualidade, reduzindo os custos de desenvolvimento (FARAHANI; RAMSIN, 2014). As entregas rápidas e frequentes de software com qualidade são determinantes para o sucesso no mercado e diferencial competitivo de uma organização desenvolvedora de software (COLOMO-PALACIOS et al., 2018; WETTINGER et al., 2015).

Para alcançar esses objetivos, vários métodos e abordagens têm sido propostos por pesquisadores e profissionais (FARAHANI; RAMSIN, 2014). Tais métodos e abordagens são catalogados pela Engenharia de Software (ES), sendo ela uma área voltada para a especificação, desenvolvimento, validação e manutenção de sistemas de software (SOMMERVILLE, 2011). Dessa maneira, a ES é composta por práticas, métodos e ferramentas que permitem a construção de sistemas de alta qualidade e de maneira apropriada (PRESSMAN, 2011), além de estar dentro do cronograma e do orçamento (SOMMERVILLE, 2011).

A ES possui e define processos, que estabelecem atividades voltadas para a produção do software, estando tais processos em contínua evolução para atender às necessidades do mercado. Os primeiros processos de desenvolvimento de software que surgiram são os chamados “tradicionais”, apresentando etapas e fases bem definidas, orientados à documentação e dependentes de formalidade na execução das atividades (PRESSMAN, 2011; SOMMERVILLE, 2011).

Esses processos deixaram de ser efetivos para o desenvolvimento de sistemas de software contemporâneo, pois tais softwares necessitam de mudanças rápidas e respostas imediatas aos usuários, devido às alterações inevitáveis e imprevisíveis de requisitos e regras de negócio (SOMMERVILLE, 2011). Nesse sentido, a abordagem dos processos “tradicionais” dificultava a execução de alterações no sistema, carecendo de flexibilidade para mudanças inesperadas e não planejadas. Isso se agrava quando considera-se que o desenvolvimento de software é uma atividade complexa caracterizada por tarefas e requisitos que apresentam

bastante variabilidade (NERUR et al., 2005).

Dessa maneira, os métodos ágeis surgiram, se consolidando a partir do Manifesto Ágil escrito em 2001 (PRESSMAN, 2011). Tais métodos consolidam processos de desenvolvimento e manutenção que dão suporte às mudanças, envolvendo o cliente no processo de criação e reduzindo a burocracia em torno do processo (SOMMERVILLE, 2011), além de auxiliarem na produção de software com qualidade, fornecendo para as organizações a capacidade de adaptação às mudanças e agilidade no desenvolvimento (SERRADOR; PINTO, 2015).

Uma extensão e continuação da jornada Ágil é a cultura DevOps (Desenvolvimento e Operações) (KIM et al., 2016). Essa cultura busca aproximar e diminuir a distância entre as equipes de desenvolvimento e operações de uma organização (WETTINGER et al., 2015), incentivando a adoção de práticas que levem à entrega contínua e integração contínua com qualidade (JABBARI et al., 2016). O DevOps possibilita que uma empresa maximize a velocidade de entrega de um produto ou serviço, desde a ideia inicial até a liberação para a produção (SHARMA, 2014).

Por se tratar de um movimento cultural, é necessário que a organização esteja alinhada às práticas e princípios de tal cultura, tendo o conhecimento suficiente para ser adotada e praticada de forma efetiva. Dessa maneira, o acesso, manutenção e reutilização do conhecimento é essencial. Considerando esse cenário, os princípios da Gestão do Conhecimento (GC) dispõem de recursos para a realização do gerenciamento formal do conhecimento (O'LEARY, 1998), capacitando as organizações a produzir, manter, adquirir e transmitir conhecimento (VASANTHAPRIYAN et al., 2015).

Este trabalho de conclusão de curso está inserido no contexto contemporâneo de processos de desenvolvimento de software. Em particular, é apresentada uma investigação do uso de recursos de GC para aperfeiçoar a adoção de DevOps, por meio de um estudo de caso executado dentro de uma empresa real de desenvolvimento de software. Adicionalmente, por meio de uma *Survey* em âmbito nacional, são mapeadas as práticas de DevOps em casos práticos de desenvolvimento e apresentados os aspectos positivos e negativos acerca do uso de DevOps e GC.

1.2 PROBLEMA DE PESQUISA

Existem problemas associados à adoção efetiva do DevOps nas organizações, tais problemas são referentes à ausência de estratégias advindas do meio acadêmico e profissional (KAMUTO; LANGERMAN, 2017), ao gerenciamento do conhecimento (WETTINGER et al.,

2015) e a falta de consciência das organizações de que o DevOps se trata de um movimento cultural (SAMARAWICKRAMA; PERERA, 2017).

De acordo com Kamuto e Langerman (2017), faltam estratégias para a adoção da cultura na comunidade acadêmica e profissional, além de existir fatores que inibem a adoção, dentre eles, a “falta de educação em torno do DevOps”, por se tratar de um tema relativamente novo. Isso significa que os profissionais envolvidos no processo de desenvolvimento não têm o conhecimento do tema ou, então, ele não está sendo transmitido de forma adequada. Dessa maneira, percebe-se um déficit em relação ao gerenciamento do conhecimento. Sobre esse déficit, Wettinger et al. (2015) discorre que o conhecimento sobre DevOps está disponível, porém não é capturado e gerenciado de forma sistemática.

Outro ponto a ser destacado, conforme Samarawickrama e Perera (2017), é o crescimento da demanda por entregas rápidas no âmbito dos negócios de software, levando as empresas a acreditarem que a solução para o problema é a aplicação do DevOps. Entretanto as organizações adotam o DevOps sem terem o conhecimento de que a prática consiste de um movimento cultural. Desse modo, identifica-se que a organização não está alinhada acerca das práticas e dos fundamentos da cultura que está sendo aplicada, prejudicando a adequada adoção do DevOps e, conseqüentemente, sua efetividade.

Diante do que foi apresentado, o problema a ser abordado no presente trabalho está associado à necessidade de esforços de pesquisa que viabilizem o compartilhamento de informações e GC a fim de cooperar com a adoção efetiva e ampla do DevOps nas equipes e organizações desenvolvedoras de sistemas. Assim, fomenta-se entregas de software mais rápidas e com qualidade.

1.3 OBJETIVOS

Este trabalho de conclusão de curso coopera com a área de ES, prioritariamente, com a área de processos de desenvolvimento de software, por meio da associação de estratégias de GC como instrumento para a viabilização da aplicação de práticas do DevOps. As seções a seguir apresentam o objetivo geral e os específicos traçados pelo estudo.

1.3.1 OBJETIVO GERAL

O objetivo geral deste trabalho é o fornecimento de um insumo de apoio à aplicação da prática DevOps em cenários práticos de desenvolvimento de software por meio do uso de recursos de GC.

1.3.2 OBJETIVOS ESPECÍFICOS

Com a finalidade de apoiar a pesquisa do presente estudo e alcançar o objetivo geral, foram elencados os seguintes objetivos específicos:

- Mapear as práticas DevOps em casos de sucesso de adoção no cenário nacional;
- Identificar potenciais benefícios e limitações na adoção de DevOps em cenários práticos;
- Estudar recursos oriundos da área de GC para apoiar a implementação de DevOps, aliviando problemas de adoção da prática;
- Aplicar o *Framework* Conceitual DOKS para avaliar o cumprimento dos principais elementos que compõe a cultura DevOps em um cenário real de desenvolvimento; e
- Avaliar a efetividade do *framework* e o uso de GC para apoiar o DevOps por meio de estudos empíricos e análises qualitativas.

1.4 JUSTIFICATIVA

No cenário corrente de desenvolvimento de software, a entrega rápida, frequente e contínua do software ao mercado, contendo novas funcionalidades e correções de defeitos é a chave para o sucesso nos negócios e o diferencial competitivo que a organização pode apresentar (WETTINGER et al., 2015; SAMARAWICKRAMA; PERERA, 2017; KIM et al., 2016).

Entretanto, conforme Kim et al. (2016) ressaltam, a maioria das organizações de software não conseguem implementar alterações que irão para a produção em minutos ou horas, sendo assim, as alterações podem levar semanas ou meses. As organizações também não são capazes de implantar centenas ou milhares de alterações em produção por dia. Em vez disso, batalham para implantar mensalmente ou até mesmo trimestralmente. Por fim, é importante destacar que o processo de implantação para a produção não costuma ser uma atividade rotineira, repleta de interrupções e problemas.

Essas organizações têm uma significativa desvantagem competitiva. Essa desvantagem poderia ser sanada com a cultura DevOps, pois conforme pesquisas (TECHNOLOGIES, 2015; PUPPET; DORA, 2017), a cultura contribui significativamente para fidelização de clientes, conquista de mercado, aumento no tempo de entrega, diminuição das taxas de falhas do sistema e aumento da frequência de implantação de código.

Porém, pelo motivo do DevOps ser um tema emergente que vem ganhando espaço (WETTINGER et al., 2015; HUSSAIN et al., 2017), a falta de um catálogo de recomendações e informações empíricas que permitam a saudável adoção de práticas DevOps é latente para diversas organizações. O estado da arte ainda é carente da identificação de boas práticas e recomendações que preencham a lacuna entre time de desenvolvimento e time de operações. Adicionalmente, *frameworks* e ferramentas específicas para a melhoria do DevOps nas organizações são necessários.

A justificativa para a realização do presente estudo está na necessidade do fornecimento de recursos para gerenciar e compartilhar o conhecimento acerca da cultura DevOps. A intenção de suprir essa necessidade está em garantir a adoção ampla e efetiva das práticas, ferramentas e abordagens da cultura pela organização, oportunizando entregas e integrações contínuas de software rápidas e com qualidade, resultando no diferencial competitivo.

1.5 ORGANIZAÇÃO DA MONOGRAFIA

Além deste Capítulo introdutório, o presente documento está organizado da seguinte forma: o Capítulo 2 realiza uma fundamentação teórica dos tópicos referentes ao tema da monografia. O Capítulo 3 descreve as avaliações empíricas realizadas para alcançar os objetivos desse trabalho acadêmico. O Capítulo 4 apresenta os resultados obtidos, bem como realiza a análise e discussões dos mesmos. O Capítulo 5 apresenta trabalhos relacionados com os tópicos de pesquisa da presente monografia. Por fim, o Capítulo 6 tece as considerações finais, perspectivas futuras e conclusões sobre o trabalho realizado.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda tópicos correlatos ao tema da pesquisa e que a apoiam, constituindo de: ES, Processos de Desenvolvimento de Software, DevOps, GC e o DOKS *Framework*.

2.1 ENGENHARIA DE SOFTWARE

O termo ES surgiu no ano de 1969, após uma ampla discussão sobre os problemas relativos à crise de software. Tais problemas estavam relacionados com o desenvolvimento de software, em geral, envolviam o atraso na entrega de grandes sistemas que não possuíam as funcionalidades requeridas pelos usuários e configuravam um elevado custo (SOMMERVILLE, 2011).

A ES é definida por Sommerville (2011) como sendo uma disciplina oriunda da engenharia tradicional cujo foco está em todos os aspectos da produção de software, incluindo as fases iniciais da especificação até o aperfeiçoamento e manutenção do sistema. Pressman (2011) destaca que a ES fornece processos, métodos e ferramentas para a construção de software de qualidade.

Tal disciplina está dividida em áreas do conhecimento, sendo estabelecidas pelo SWEBOK (*Software Engineering Body of Knowledge*). Este, por sua vez, é um guia criado pela IEEE (*Institute of Electrical and Electronics Engineers*) com o objetivo de estabelecer uma base de conhecimento para o campo da ES, caracterizando os limites de tal disciplina (SWEBOK, 2014), tais áreas são: (1) Requisitos de Software; (2) Projeto de Software; (3) Construção de Software; (4) Teste de Software; (5) Manutenção de Software; (6) Gerência de Configuração de Software; (7) Gerenciamento de Engenharia de Software; (8) Processo de Engenharia de Software; (9) Modelos e Métodos de Engenharia de Software; (10) Qualidade de Software; (11) Prática Profissional de Engenharia de Software; (12) Economia de Engenharia de Software; (13) Fundamentos de Computação; (14) Fundamentos Matemáticos; e (15) Fundamentos de Engenharia.

As áreas de conhecimento dão uma cobertura sobre as atividades técnicas e gerenciais que englobam o ciclo de vida do software, sendo elas executadas durante a aquisição, desenvolvimento, manutenção e desativação do software (SWEBOK, 2014). Para a concepção de um software de qualidade, o Swebok (2014) estabelece as áreas relativas aos processos de ES e os modelos e métodos de desenvolvimento. Tais áreas serão abordadas nas seções a seguir, bem como contemplando um histórico da evolução na forma de se desenvolver software.

2.2 PROCESSOS DE DESENVOLVIMENTO DE SOFTWARE

Os processos podem ser definidos como um conjunto de atividades, ações e tarefas relacionadas que são realizadas para a criação e produção de um produto de software (SOMMERVILLE, 2011; PRESSMAN, 2011).

Sommerville (2011) especifica algumas atividades fundamentais de um processo: *(i)* Especificação do software (requisitos e restrições); *(ii)* Projeto e implementação de software (construir o sistema para atender às especificações); *(iii)* Validação do software (validar se atende às demandas dos clientes); e *(iv)* Evolução do software (evoluir conforme necessidades dos clientes).

Os processos são executados quando algum produto de software precisa ser criado, eles definem etapas que auxiliam na criação de um resultado de qualidade, além de providenciar controle e organização das atividades a serem realizadas (PRESSMAN, 2011).

Sommerville (2011) aponta que não existe um processo ideal e que há espaço para melhorias no processo dentro das organizações. O autor ainda ressalta que os processos têm evoluído de modo a tirarem o melhor proveito das capacidades das pessoas e de acordo com as características do sistema a ser construído. Dessa forma, na sequência são apresentados os processos e metodologias tradicionais de desenvolvimento e os métodos ágeis, que representam o início e a evolução na maneira de se desenvolver software.

2.2.1 PROCESSOS E METODOLOGIAS DE DESENVOLVIMENTO TRADICIONAIS

Os primeiros processos, chamados de “tradicionais”, foram propostos para organizar o caos que existia na área de desenvolvimento de software (PRESSMAN, 2011). O autor ainda destaca que eles contribuíram para a estruturação da ES.

Os processos tradicionais têm por característica fases distintas e bem definidas, tais como levantamento e análise de requisitos, modelagem do software, desenvolvimento, testes e

entrega (SOMMERVILLE, 2011). Além disso, são orientados à documentação e dependentes de formalidade na execução das atividades (PRESSMAN, 2011; SOMMERVILLE, 2011).

O primeiro modelo do processo de desenvolvimento de software é o Modelo Cascata (Figura 1), algumas vezes chamado de ciclo de vida clássico (PRESSMAN, 2011), esse modelo sugere uma abordagem sequencial e sistemática. Dessa maneira, possui o encadeamento entre uma fase e outra, na qual o estágio seguinte não deve ser iniciado antes do término da fase anterior e o resultado de cada estágio é a aprovação de um ou mais documentos (SOMMERVILLE, 2011). As principais fases e atividades desse modelo são: Análise e definição de requisitos, Projeto de sistema e software, Implementação e teste unitário, Integração e testes de sistema e Operação e Manutenção.

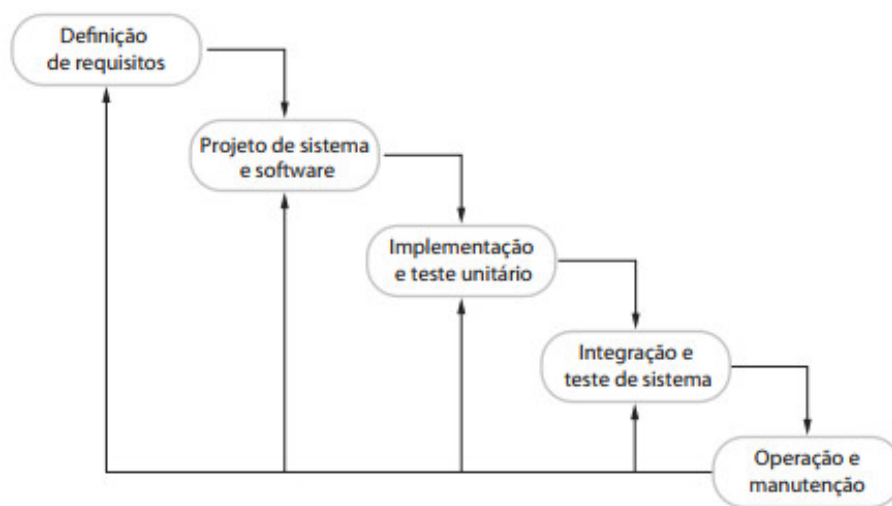


Figura 1: Representação do Modelo em Cascata e suas atividades sequenciais.

Fonte: Sommerville (2011)

Uma variação do Modelo Cascata é o Modelo em V (Figura 2), que relaciona as ações de garantia de qualidade e as ações associadas à comunicação, modelagem e atividades de construção iniciais (PRESSMAN, 2011). A equipe de software desce em direção ao lado esquerdo do V, realizando as atividades para o desenvolvimento do software, uma vez finalizado a equipe se desloca para cima, no lado direito do V, que consiste na realização das atividades de testes, verificação e validação. Não existe nenhuma diferença fundamental entre as etapas do Modelo Cascata e do Modelo em V, o Modelo em V apenas fornece uma maneira de visualizar a aplicação das atividades de verificação e validação que são realizados no modelo anterior (PRESSMAN, 2011).

O desenvolvimento de software tem um ritmo acelerado e está sujeito à mudanças intermináveis, sendo o Modelo Cascata inapropriado para tal trabalho (PRESSMAN, 2011),

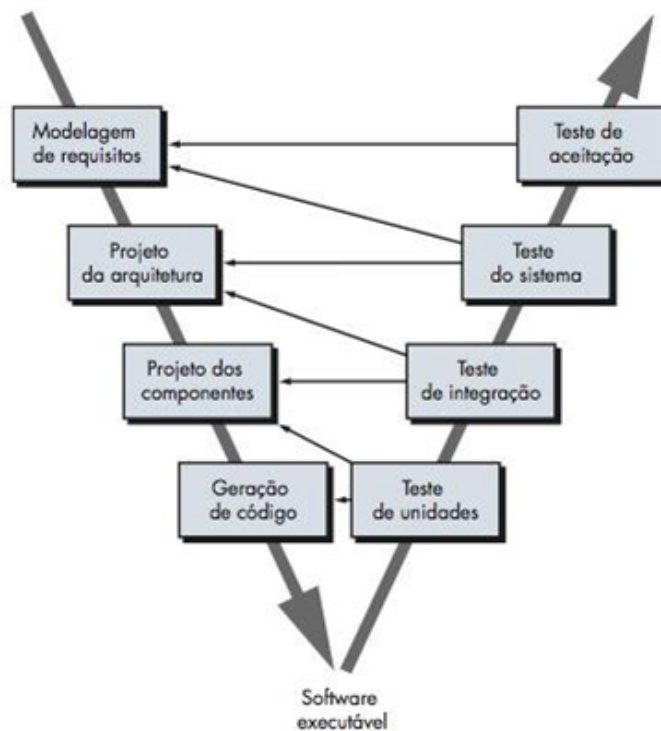


Figura 2: Representação do Modelo em V e suas atividades.

Fonte: Pressman (2011)

indica-se a utilização de tal modelo para situações nas quais os requisitos são fixos e bem definidos, e o trabalho é realizado de maneira linear (PRESSMAN, 2011; SOMMERVILLE, 2011). Entretanto, em algumas situações os requisitos iniciais de software são razoavelmente bem definidos, podendo ser interessante o fornecimento rápido de uma versão inicial do software para ser avaliado pelos usuários (PRESSMAN, 2011; SOMMERVILLE, 2011).

Desse modo, pode optar-se a desenvolver o software de forma incremental, para tanto existe a abordagem do Modelo Incremental (Figura 3). Esse modelo segue uma abordagem escalonada de desenvolvimento e visa gerar incrementos gradativos do software (PRESSMAN, 2011), ou seja, executa-se as etapas do processo de desenvolvimento, ao final entrega-se uma pequena parte do software e o ciclo se repete até a entrega completa. O desenvolvimento incremental é baseado na ideia de desenvolver uma implementação inicial, expô-la aos comentários dos usuários e continuar por meio da criação de várias versões até o sistema adequado ser desenvolvido (SOMMERVILLE, 2011).

Outra abordagem tradicional de desenvolvimento de software são os modelos de processos evolucionários, tais modelos são iterativos e apresentam características que possibilitam desenvolver versões cada vez mais completas do software (PRESSMAN, 2011). Exemplos dessa abordagem são o Prototipagem e o Modelo Espiral.

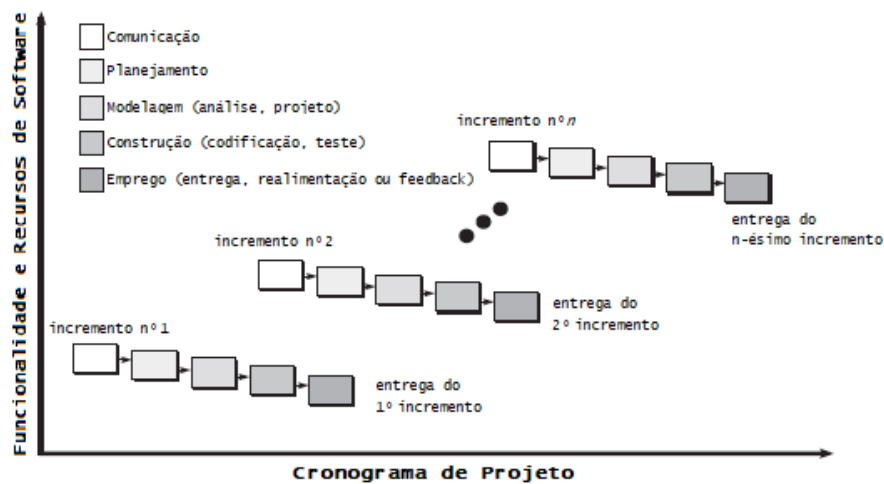


Figura 3: Representação do Modelo Incremental.

Fonte: Adptado de Pressman (2011)

A Prototipagem (Figura 4) pode ser utilizada quando o cliente define os objetivos gerais do software, mas não identifica detalhadamente os requisitos. Nessa abordagem é construído um protótipo do sistema, com objetivo de demonstrar os conceitos do software, representar aspectos visíveis aos usuários e servindo como forma de verificação da viabilidade do projeto e aprimoramento dos requisitos (PRESSMAN, 2011; SOMMERVILLE, 2011).

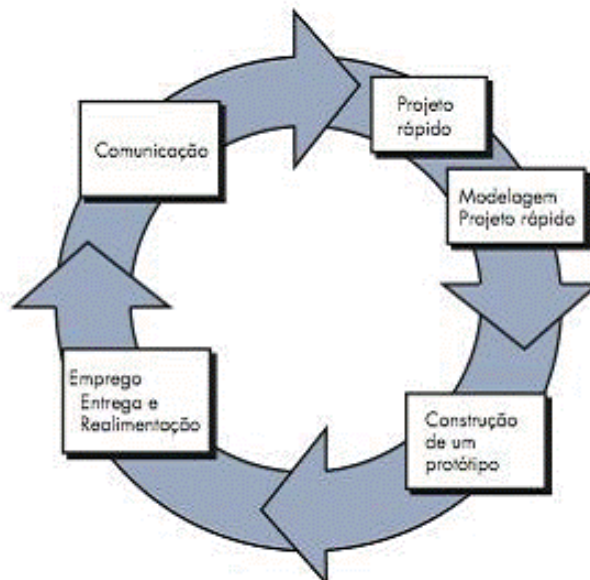


Figura 4: Representação do fluxo da Prototipagem.

Fonte: Pressman (2011)

O Modelo Espiral (Figura 5) agrega características da prototipação com aspectos

sistemáticos e controlados do Modelo Cascata (PRESSMAN, 2011). O processo de software é representado como uma espiral e cada volta da espiral representa uma fase dentro do processo de software (SOMMERVILLE, 2011). Sendo assim, o software será desenvolvido em uma série de versões evolucionárias.

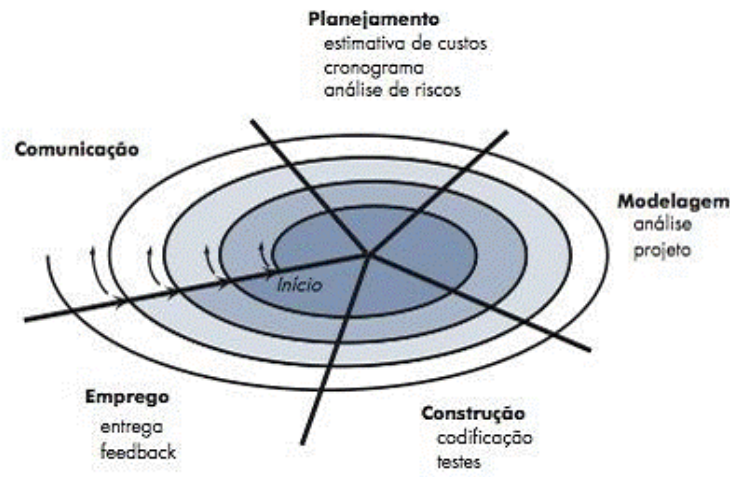


Figura 5: Representação do Modelo Espiral.

Fonte: Pressman (2011)

Contudo, os processos tradicionais descritos não são adequados para o desenvolvimento efetivo de software contemporâneo, que possuem como características mudanças e entregas rápidas (SOMMERVILLE, 2011). Isso ocorre pois tais processos são burocráticos, dependentes de formalidades, dão importância à documentação e criação de artefatos intermediários do software (PRESSMAN, 2011; SOMMERVILLE, 2011) e essas características dificultam alterações no sistema ocasionadas por mudanças de requisitos ou pela correção de defeitos no meio do processo.

Por consequência, surgiram os Métodos Ágeis em resposta aos métodos "tradicionais", que não eram aptos para satisfazer as dinâmicas do mercado (CRUZ, 2015) e não possibilitavam o desenvolvimento rápido com capacidade de mudanças (SOMMERVILLE, 2011).

2.2.2 METODOLOGIAS ÁGEIS

Os Métodos Ágeis surgiram no final da década de 1990, propondo uma nova abordagem para processos de desenvolvimento de software (SOMMERVILLE, 2011). Esses destinam-se a entregar o software rapidamente ao cliente, num contexto suscetível às mudanças, além de priorizar a comunicação entre desenvolvedores, clientes e qualquer outro interessado do projeto (*stakeholders*) (PRESSMAN, 2011; SOMMERVILLE, 2011).

No ano de 2001, um grupo de dezessete especialistas e notáveis profissionais da área de software se reuniram e criaram o que hoje é chamado de Manifesto Ágil, que expõe a filosofia por trás dos métodos ágeis (SOMMERVILLE, 2011). O Manifesto (BECK et al., 2001) incentiva a descoberta e elaboração de maneiras melhores de desenvolver software. Para tanto, são definidos quatro valores que guiam o desenvolvimento ágil de software:

- Indivíduos e interações mais que processos e ferramentas;
- Software em funcionamento mais que documentação abrangente;
- Colaboração com o cliente mais que negociação de contratos; e
- Responder às mudanças mais que seguir um plano.

Nerur et al. (2005) ressaltam que os métodos ágeis são caracterizados por ciclos iterativos e curtos de desenvolvimento, compostos de períodos de reflexão e introspecção. Adicionalmente, os métodos ágeis apoiam as tomadas de decisão de forma colaborativa, almejam o *feedback* e mudanças rápidas e contemplam a integração contínua das alterações no sistema em desenvolvimento. Dentre as principais metodologias ágeis, destacam-se: Scrum e XP (*Extreme Programming*) (SOMMERVILLE, 2011; PRESSMAN, 2011).

O Scrum (Figura 6) é considerado um *framework* para desenvolver, entregar e manter produtos complexos (SCHWABER; SUTHERLAND, 2017). O Scrum é pautado na realização de Sprints, período de 15 dias ou um mês no qual são realizadas as seguintes cerimônias, conforme Schwaber e Sutherland (2017): (i) Planejamento da Sprint: define-se o trabalho a ser realizado durante a Sprint; (ii) Reuniões diárias: nessas reuniões o time planeja o trabalho para as próximas 24 horas; (iii) Trabalho de desenvolvimento: atividade para desenvolver o que foi estipulado para a Sprint; (iv) Revisão da Sprint: realizada ao final da Sprint, serve para inspecionar o que foi realizado e adaptar os requisitos; e (v) Retrospectiva da Sprint: cerimônia na qual a equipe faz uma avaliação de si mesma e planeja ações de melhorias para a próxima Sprint.

Em particular, o Scrum também possui artefatos e o time Scrum (SCHWABER; SUTHERLAND, 2017). Dentre os artefatos, pode-se destacar: (i) *Product Backlog*, em português Backlog do Produto, que consiste de uma lista com prioridades dos requisitos ou funcionalidades do projeto (PRESSMAN, 2011); e (ii) *Sprint Backlog*, que contempla um conjunto de itens do Backlog do Produto para serem realizados durante a Sprint. O Time Scrum é formado pelos seguintes papéis, conforme Schwaber e Sutherland (2017): (i) *Product Owner*, em português Dono do Produto, é responsável por gerenciar o *Product Backlog* e maximizar

o valor do produto entregue; (ii) Time de Desenvolvimento, conjunto de profissionais que realizam o trabalho de entregar um incremento do produto ao final da Sprint; e (iii) Scrum Master, responsável por guiar o time na aplicação do Scrum, acompanhar o desenvolvimento e resolver impedimentos do time.

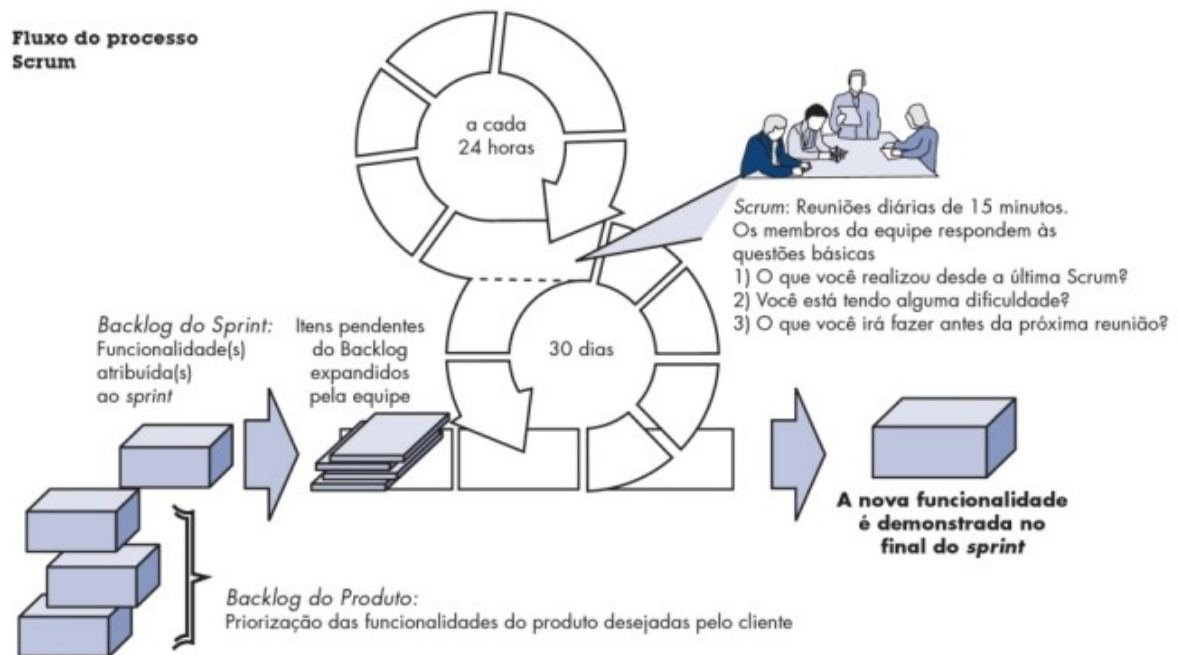


Figura 6: Representação do Framework Scrum.

Fonte: Pressman (2011)

O XP (Figura 7) é uma abordagem para impulsionar boas práticas de desenvolvimento (SOMMERVILLE, 2011). Nela, os requisitos estão apresentados como cenários, os programadores trabalham em pares e desenvolvem testes para cada tarefa antes de desenvolverem e ao integrarem o que foi desenvolvido, todos os testes devem ser executados com sucesso (PRESSMAN, 2011; SOMMERVILLE, 2011). Conforme Sommerville (2011), o XP envolve práticas que refletem os princípios dos métodos ágeis, tal como: o desenvolvimento é incremental, há o envolvimento com o cliente e mudanças são aceitas no decorrer do processo.

A adoção das metodologias que utilizam os valores ágeis torna possível a entrega de produtos mais coerentes e relevantes aos clientes, como visto no trabalho de Machado (2017). As metodologias ágeis também são apontadas como abordagens que resolvem os problemas encontrados nos métodos “tradicionais” de desenvolvimento, provendo eficiência, satisfação das partes interessadas e percepção de desempenho do projeto (SERRADOR; PINTO, 2015).

Devido à adoção de princípios e práticas ágeis, o tempo necessário para desenvolver

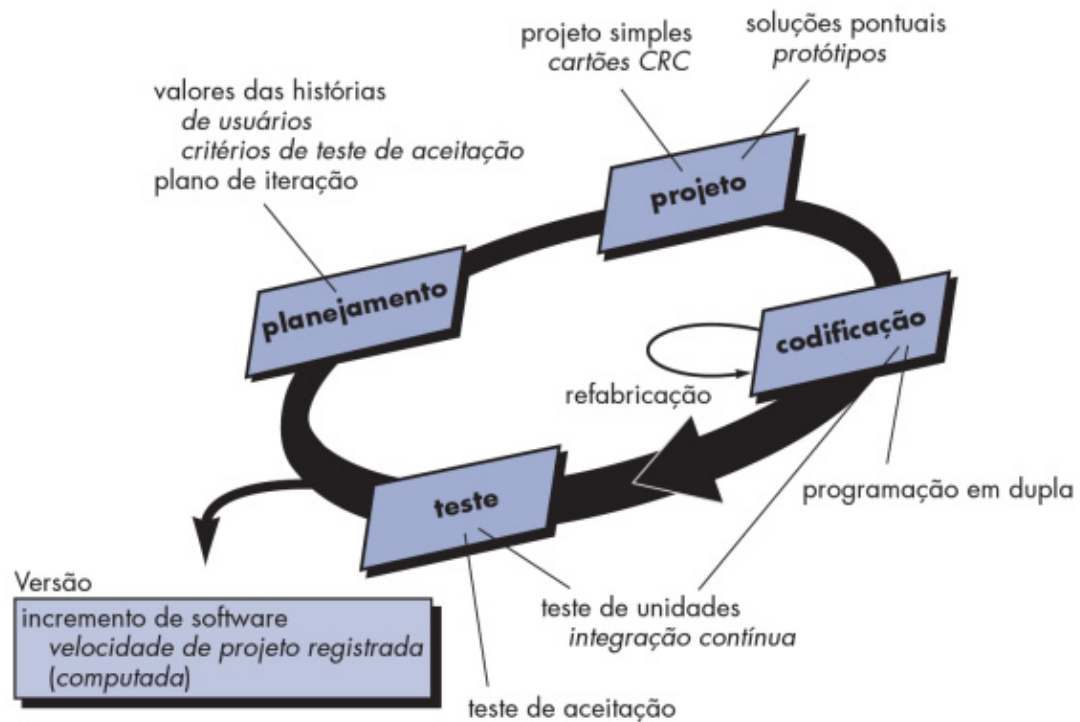


Figura 7: Representação do XP.

Fonte: Pressman (2011)

novas funcionalidades diminuiu para semanas ou meses (KIM et al., 2016). Contudo, a etapa referente à entrega e implantação do produto para o cliente (no chamado ambiente de produção), conforme Kim et al. (2016), ainda exige tempo e maior atenção, devido aos frequentes problemas e complicações durante a execução dessa etapa. Dessa maneira, uma extensão dos métodos ágeis que auxilia na implantação de novas funcionalidades para a produção com a aproximação das equipes de desenvolvimento e operações, utilizando princípios ágeis, é o DevOps (DAVIS; DANIELS, 2016; JABBARI et al., 2016; KIM et al., 2016).

2.3 DEVOPS

DevOps é visto como um movimento cultural que tem por intuito diminuir a distância entre o time de desenvolvimento e o time de operações (DAVIS; DANIELS, 2016; WETTINGER et al., 2015). O time de desenvolvimento é responsável por criar as aplicações, adicionar funcionalidades e corrigir defeitos, enquanto o time de operações é responsável por cuidar das aplicações em produção, zelando pela estabilidade (SATO, 2017).

Nas organizações tradicionais, os times são criados de acordo com trabalho que

desempenham, em que cada departamento (silo) tem seus próprios objetivos e metas (KAMUTO; LANGERMAN, 2017). Nessa perspectiva, de acordo com Sato (2017) e Sharma (2017) gera-se um conflito entre os objetivos, pois os desenvolvedores entregam as funcionalidades e correções, desejando que essas entregas estejam o mais rápido possível em produção, porém a equipe de operações evita mudanças, visto que pode afetar a estabilidade do sistema.

Portanto, o objetivo do DevOps é remover as barreiras existentes entre esses times, aproximando-os e criando um canal de comunicação e colaboração mútuo, levando a uma maior transparência e confiança entre as equipes e trabalhando em direção às metas organizacionais compartilhadas (DAVIS; DANIELS, 2016). Além de melhorar as entregas de software, agregando qualidade, segurança e capacidade de obter *feedback* rápido sobre os produtos.

DevOps compartilha muitas características com os métodos ágeis, tais como foco nos indivíduos, interações, colaboração, equipes pequenas e entregas de qualidade ao cliente, assim, muitas práticas e princípios do DevOps são compatíveis com os Ágeis. Kim et al. (2016) descreve que o DevOps é uma continuação do movimento ágil.

2.3.1 HISTÓRIA DO DEVOPS

A história do DevOps começa em 2008, quando Patrick Debois, gerente de projetos e especialista em desenvolvimento, propôs a discussão de métodos para solucionar os conflitos entre as áreas de desenvolvimento e operações dentro de uma organização (DAVIS; DANIELS, 2016; KIM et al., 2016), por meio da apresentação e publicação do artigo intitulado “*Agile Infrastructure and Operations: How Infra-gile are You?*” (DEBOIS, 2008).

No ano seguinte, na *Velocity Conference*¹, dois profissionais que trabalhavam no Flickr² apresentaram um seminário entitulado “*10+ Deploys per Day: Dev and Ops Cooperation at Flickr*” (DAVIS; DANIELS, 2016) e Debois percebeu que se iniciava um pensamento referente à integração entre a equipe de desenvolvimento e a de operações, o que o motivou a organizar o primeiro DevOpsDay (conferência que trata de tópicos relacionados ao DevOps). Assim, deu origem ao termo DevOps (KIM et al., 2016), que é formado a partir das palavras “desenvolvimento” (do inglês *development*) e “operações” (do inglês *operations*).

Apesar de ser um movimento relativamente recente, grandes empresas já usufruem

¹É um evento para profissionais de TI, no qual são realizadas apresentações e workshops com especialistas da área de desenvolvimento de software. O objetivo é capacitar os profissionais para lidar com sistemas e aplicativos modernos. Link do evento: <https://conferences.oreilly.com/velocity>.

²Flickr - <https://www.flickr.com/>

dessa cultura, tal como Netflix³, Facebook⁴, Amazon⁵ e Google⁶ (KIM et al., 2016; SAVOR et al., 2016; HUMBLE; MOLESKY, 2011), propiciando a entrega e implantação contínua dos seus respectivos sistemas (KIM et al., 2016; SAVOR et al., 2016).

2.3.2 ASPECTOS CONCEITUAIS DE DEVOPS

Na literatura encontra-se alguns trabalhos que fazem referências aos conceitos e princípios que auxiliam no entendimento, compreensão e aplicação do DevOps, definindo aspectos e valores que são importantes para quem deseja adotar o DevOps. A seguir, são apresentadas a descrição dos conceitos e princípios encontrados.

2.3.2.1 PILARES DO DEVOPS

Davis e Daniels (2016) descreve quatro pilares para a implementação efetiva do Devops, representados na Figura 8, que juntos proveem mudanças e soluções de problemas culturais e técnicos que podem impactar no desenvolvimento do software, são eles:

- Colaboração (*Collaboration*): a colaboração é o processo de construir um resultado específico por meio do apoio às interações e a entrada de várias pessoas. Um princípio que moldou o movimento DevOps foi a cooperação das equipes de desenvolvimento e operações de software. Antes que as equipes trabalhem juntas e com sucesso, é necessário que os indivíduos de uma equipe consigam trabalhar juntos, pois uma equipe que não funciona bem no nível individual, dificilmente terá sucesso ao trabalhar no nível de interação;
- Afinidade (*Affinity*): a afinidade é o processo de construir relacionamentos interdependentes e fortes entre os indivíduos e os times. Levando em conta as metas pessoais, ao mesmo tempo em que mantém em mente os objetivos organizacionais compartilhados e promove a empatia e o aprendizado entre diferentes grupos de pessoas. A afinidade também pode ser aplicada entre organizações, permitindo que as empresas compartilhem experiências e aprendam umas com as outras à medida que produzem conhecimento cultural e técnico dentro da organização; desenvolver e manter um ambiente de trabalho aberto, confiável e comunicativo é importante tanto para as equipes que trabalham unidas quanto para os indivíduos. O compartilhamento de experiências e

³Netflix - <https://www.netflix.com/>

⁴Facebook - <https://www.facebook.com/>

⁵Amazon - <https://www.amazon.com.br/>

⁶Google - <https://www.google.com.br/>

ideias auxilia na manutenção de um ambiente de inovação e confiança, sendo fundamental para a cultura DevOps;

- Ferramentas (*Tools*): as ferramentas funcionam como aceleradoras e apoiadoras da cultura DevOps, quando escolhidas de forma correta e que se encaixam nos processos da organização. É necessário escolher a ferramenta certa para tratar os problemas reais e existentes que afetam realmente a execução do trabalho. É importante analisar qual ferramenta atende às necessidades para evitar problemas futuros e que afetem a cultura. Ferramentas que dão suporte para as atividades dentro do processo de desenvolvimento, tais como gerenciamento de versão, automação de testes e *build*⁷ e monitoramento são importantes para a construção da cultura DevOps; e
- Dimensionamento (*Scaling*): o dimensionamento trata sobre a evolução, o crescimento e o avanço da organização como um todo ao longo de todo o seu ciclo de vida e os problemas que podem ocorrer com o crescimento ou até retrocesso da empresa. Uma empresa de sucesso deve saber como crescer ou retroceder quando necessário e saber usar de maneira correta os três pilares anteriores de acordo com o tamanho de sua organização.

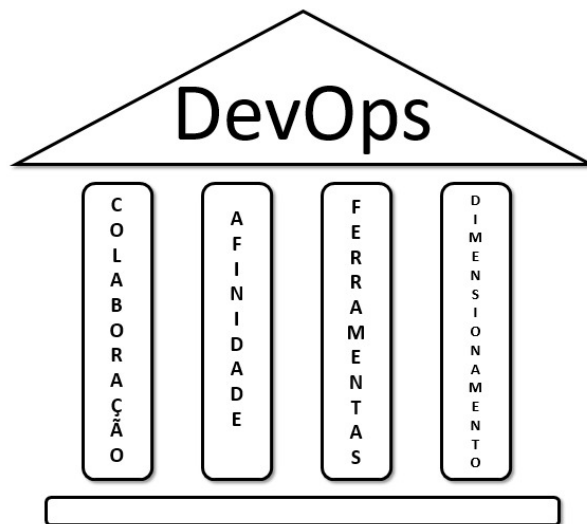


Figura 8: Pilares do DevOps.

Fonte: Autoria própria

⁷*Build* é a etapa na qual o código é compilado para criar e testar a unidade dos binários a serem implantados (SHARMA, 2014).

2.3.2.2 CULTURA, AUTOMAÇÃO, MEDIÇÃO E COMPARTILHAMENTO

O *framework* CAMS foi criado por John Willis and Damon Edward, sendo um conjunto de valores que descrevem o DevOps e dão base para muitos profissionais da área (PERERA et al., 2017), sua representação é vista na Figura 9. É um acrônimo para as palavras Cultura (*Culture*), Automação (*Automation*), Medição (*Measurement*) e Compartilhamento (*Sharing*), sendo definidas como:

- Cultura: é o aspecto mais importante, no qual se configura um ambiente com as equipes de operações e desenvolvimento trabalhando juntas e compartilhando responsabilidades, além de participarem de reuniões e cerimônias que fazem parte do processo de desenvolvimento, a fim de facilitar a troca de experiência e conhecimento (HUMBLE; MOLESKY, 2011; PERERA et al., 2017);
- Automação: trata da automatização de processos rotineiros e repetitivos, como testes, versionamento, gestão de configuração, *build*, *deploy*⁸, monitoração. São necessárias ferramentas, pessoas e processos para se criar uma estrutura de automação para o DevOps (PERERA et al., 2017);
- Medição: refere-se ao monitoramento de métricas, tais como, frequência de implantação, níveis de desempenho, taxas de falhas, entre outras, devendo ser transparentes, abertas e acessíveis (HUMBLE; MOLESKY, 2011; PERERA et al., 2017). O DevOps bem adotado deve medir tudo o que for possível, pois com medições é possível melhorar; e
- Compartilhamento: compartilhar conhecimento, experiências, ideias, problemas, ferramentas e descobertas é o suporte para a colaboração dentro do DevOps, que garante uma construção eficaz da cultura (HUMBLE; MOLESKY, 2011; PERERA et al., 2017).

2.3.3 PRÁTICAS DE DEVOPS

O DevOps é cercado de práticas que dão suporte à adoção da cultura. Tais práticas estão principalmente relacionadas com enviar mudanças o quanto antes para a produção, reduzir ou eliminar erros durante a implantação e encontrar e reparar falhas com rapidez no sistema (BASS, 2018).

Os relatórios referentes ao estado e situação do DevOps nos anos de 2016 e 2017 (PUPPET; DORA, 2016, 2017) apresentam o aumento das evidências que as práticas de DevOps

⁸*Deploy* é o processo de implantação do sistema, podendo ser em ambientes de testes e em produção (NIELSEN et al., 2017).



Figura 9: Representação do *framework* CAMS.

Fonte: Autoria própria

ajudam a entregar o software mais rapidamente, com mais confiabilidade e com menos erros, além de melhorar a cultura organizacional e o engajamento dos funcionários das organizações.

Conforme os trabalhos de Jabbari et al. (2016) e Davis e Daniels (2016), e os relatórios sobre o estado do DevOps dos anos de 2015, 2016 e 2017 (PUPPET; DORA, 2015, 2016, 2017), as principais práticas do DevOps são:

- **Controle de Versão:** essa prática consiste em registrar e gerenciar alterações e versões de artefatos que são gerados durante o desenvolvimento (DAVIS; DANIELS, 2016; PRESSMAN, 2011). Tais artefatos podem ser código-fonte, diagramas, documento de requisitos, entre outros. Essa prática minimiza os riscos, pois assegura uma maneira de reverter objetos em produção para versões anteriores (DAVIS; DANIELS, 2016);
- **Testes automatizados:** consiste na automatização dos testes realizados de forma manual. Tem por objetivo fazer com que a ferramenta ou sistema execute o trabalho manual e repetitivo feito pelos testadores, visto que a atividade de teste consome um tempo significativo no desenvolvimento de software (VINCENZI et al., 2018);
- **Integração Contínua:** é uma prática que visa a integração dos códigos desenvolvidos de forma frequente, isso é, os desenvolvedores integram seu trabalho com frequência, geralmente diária, que passarão por testes, servindo para encontrar e investigar os erros rapidamente e corrigi-los de forma imediata (DAVIS; DANIELS, 2016; HUMBLE; FARLEY, 2010; STAHL et al., 2017);
- **Entrega Contínua:** essa prática é considerada um passo à frente da Integração Contínua (DAVIS; DANIELS, 2016), que visa garantir a aptidão das novas mudanças que foram

feitas de serem colocadas no ambiente de produção, passando por testes e avaliações que irão aferir se determinada mudança será implantada (DAVIS; DANIELS, 2016; HUMBLE; FARLEY, 2010). Essa prática normalmente envolve atividades como análise de código, teste de aceitação, verificação de requisitos, etc (STAHL et al., 2017);

- **Implantação Contínua:** essa prática está vinculada à equipe de operações, compreendendo o processo de implantar as mudanças, advindas da entrega contínua, no ambiente de produção regularmente, levando o produto ao cliente mais rapidamente, tornando as alterações disponíveis para os usuários (DAVIS; DANIELS, 2016; STAHL et al., 2017); e
- **Monitoramento:** é uma prática que propõe-se a observar e analisar o comportamento do sistema. Se está de acordo com o desejado, visando o bom desempenho do mesmo (SCHLOSSNAGLE, 2018). Fornecer mecanismos para monitorar o desempenho em todos os ambientes, desde a integração contínua e ambientes de teste até a implantação na produção, permite que o desempenho seja bem avaliado (GOTTESHEIM, 2015).

As práticas do DevOps, por vezes, são sobrepostas e uma dá suporte à outra (VADAPALLI, 2017), sendo possível suas realizações por conta da automação das atividades que compõe tais práticas. Isso impacta diretamente na agilidade em relação ao cumprimento da atividade e elimina possíveis erros humanos (PUPPET; DORA, 2017). Para o sucesso na realização das práticas estão disponíveis uma gama de ferramentas e serviços. Um catálogo dessas ferramentas está disponível no formato de uma tabela periódica, como é possível verificar no Anexo A da presente monografia, separando-as de acordo com o tipo de atividade que visam executar.

2.4 GESTÃO DO CONHECIMENTO

A GC, de acordo com Nonaka e Teece (2001), representa a criação de um ambiente de aprendizagem para apoiar a criação e transferência de conhecimento, além de usar e reutilizar o conhecimento pessoal e organizacional. O objetivo da área, conforme Vasanthapriyan et al. (2015), é a criação e manutenção de um sistema capaz de produzir, manter, aprimorar, adquirir e transmitir a base de conhecimento da organização.

O conhecimento é a interpretação da informação dentro de um contexto, sendo o resultado da percepção, aprendizagem e raciocínio (VASANTHAPRIYAN et al., 2015). O conhecimento são todas as informações processadas, fornecendo base para a tomada de decisões e ações. Existem dois tipos de conhecimento: o explícito e o tácito (NONAKA et al., 2000).

O conhecimento explícito representa o conhecimento racional e técnico, que é possível documentar e armazenar, sendo transmitido e usado facilmente por vários indivíduos (NONAKA et al., 2000). Ele pode ser armazenado em documentos, banco de dados, mensagens ou códigos (VASANTHAPRIYAN et al., 2015).

O conhecimento tácito, por outro lado, é subjetivo e baseado na experiência, não podendo ser documentado e normalmente é inerente à pessoa (SOUZA et al., 2014). Ele abrange fatores intangíveis, tais como crenças, valores e intuição (NONAKA; TAKEUCHI, 1995), não podendo ser gerenciado e ensinado da mesma maneira que o explícito (VASANTHAPRIYAN et al., 2015).

A conversão do conhecimento tácito para o explícito e a distribuição de tal conhecimento proporciona grande valor para as organizações (NONAKA; TAKEUCHI, 1995), a conversão e distribuição do conhecimento são essenciais para a organização sustentar uma vantagem competitiva (ALAVI; LEIDNER, 1999).

No que tange à criação do conhecimento, Nonaka e Takeuchi (1995) definem um processo para a conversão entre conhecimento tácito e explícito, sendo chamado de SECI ou espiral do conhecimento. Esse processo é formado por quatro interações de conversão, que formam o acrônimo que dá nome ao processo, chamadas de: Socialização, Externalização, Combinação e Internalização. A representação desse processo pode ser elucidada na Figura 10.



Figura 10: Processo SECI para conversão e criação de conhecimento.

Fonte: Adaptado de Nonaka e Takeuchi (1995)

A “Socialização”, conforme Nonaka et al. (2000), é o processo de converter novos

conhecimentos tácitos por meio de experiências compartilhadas. As empresas, geralmente, adquirem e aproveitam o conhecimento tácito a partir da interação com clientes e fornecedores.

A “Externalização” converte conhecimento tácito para explícito, quando o conhecimento tácito é exposto, permite-se que seja compartilhado para outros indivíduos e se torne a base do novo conhecimento (NONAKA et al., 2000).

A “Combinação” é o processo de conversão do conhecimento explícito em conjuntos mais complexos e sistemáticos de conhecimento explícito. O conhecimento explícito é coletado dentro ou fora da organização e depois combinado, editado ou processado para formar novos conhecimentos, que posteriormente será disseminado (NONAKA et al., 2000).

A “Internalização” incorpora conhecimento explícito ao tácito, o conhecimento explícito criado é compartilhado por toda a organização e convertido em conhecimento tácito pelos indivíduos, se tornando um recurso valioso, pois pode desencadear uma nova espiral de criação de conhecimento quando é compartilhado com os outros por meio da “Socialização” (NONAKA et al., 2000).

Com o intuito de concretizar o objetivo da GC, que inicia na criação de conhecimento até a sua transmissão e compartilhamento, existem técnicas e ferramentas que dão apoio para a realização das atividades. Na Seção 2.4.1 são introduzidas algumas dessas técnicas e ferramentas.

2.4.1 TÉCNICAS E FERRAMENTAS DE GESTÃO DE CONHECIMENTO

A GC, de acordo com Vasanthapriyan et al. (2015), envolve o gerenciamento formal dos recursos de conhecimento, com o intuito de facilitar o acesso e a reutilização do mesmo. Geralmente, ferramentas e técnicas, juntamente com a tecnologia, são usadas para favorecer a comunicação, colaboração e gerenciamento do conteúdo, para uma melhor captura, compartilhamento, disseminação e aplicação do conhecimento.

Com base nos estudos de Vasanthapriyan et al. (2015), Razzak e Ahmed (2014) e Sharma et al. (2012), as principais técnicas e ferramentas para auxiliar nas atividades de GC no desenvolvimento de software e suas aplicações em relação ao conhecimento estão expostas no Quadro 1.

Quadro 1: Principais ferramentas e técnicas para a GC e suas aplicações.

Ferramentas e Técnicas	Contribuição ao conhecimento
Repositórios (Repositórios de dados, multimídia, melhores práticas e documentos)	Compartilhar o conhecimento
Wiki	Compartilhar o conhecimento
Teleconferência, videoconferência, conferências online	Compartilhar o conhecimento
Workshops e seminários	Criar e compartilhar o conhecimento
Interação com o cliente	Criar o conhecimento
Software colaborativo (mensageiros instantâneos, e-mail)	Compartilhar o conhecimento

Fonte: Autoria própria

2.5 DOKS FRAMEWORK

O *DevOps Knowledge Sharing Framework* (DOKS), representado na Figura 11, é proposto por Nielsen et al. (2017) e possui o objetivo de auxiliar na avaliação do cumprimento dos principais elementos que compõem a cultura DevOps, dando suporte às empresas ou equipes que desejam adotar a cultura. O *framework* elenca os princípios e práticas do DevOps, que os autores identificaram como as mais recomendadas e abrangentes. O DOKS está dividido em três seções que são: (i) Princípios, (ii) Práticas e Ferramentas e (iii) GC.

Os Princípios do DevOps expostos pelo DOKS são Agilidade, Colaboração e Integração. O princípio “Agilidade” descrito pelos autores está relacionado à contínua e frequente entrega de software, com um estreito relacionamento entre o time de negócios e de desenvolvimento, além de reflexões e melhorias no desenvolvimento. A “Colaboração” está relacionada com o respeito, confiança e abertura entre os membros da equipe, em conjunto com a divisão de responsabilidades. E o princípio de “Integração” é associado às práticas e ferramentas, envolvendo a automação, o uso de boas práticas e serviços na nuvem.

Em relação às Práticas e Ferramentas, o DOKS divide as etapas de desenvolvimento e entrega de software que são realizados pelos desenvolvedores (“Medir e Planejar” e “Desenvolver e Testar”) e pela equipe de operações (“Lançar e Implantar” e “Monitorar e Otimizar”), elencando as respectivas práticas de cada etapa.

Na GC, o DOKS utiliza como base o modelo SECI, com o fim de analisar e descrever as práticas de compartilhamento de conhecimento entre as equipes de desenvolvimento e operações em organizações de desenvolvimento de software. O modelo SECI é utilizado, pois de acordo com os autores, pode aumentar a conscientização sobre as diferentes maneiras de compartilhar o conhecimento, este sendo um importante aspecto do DevOps.

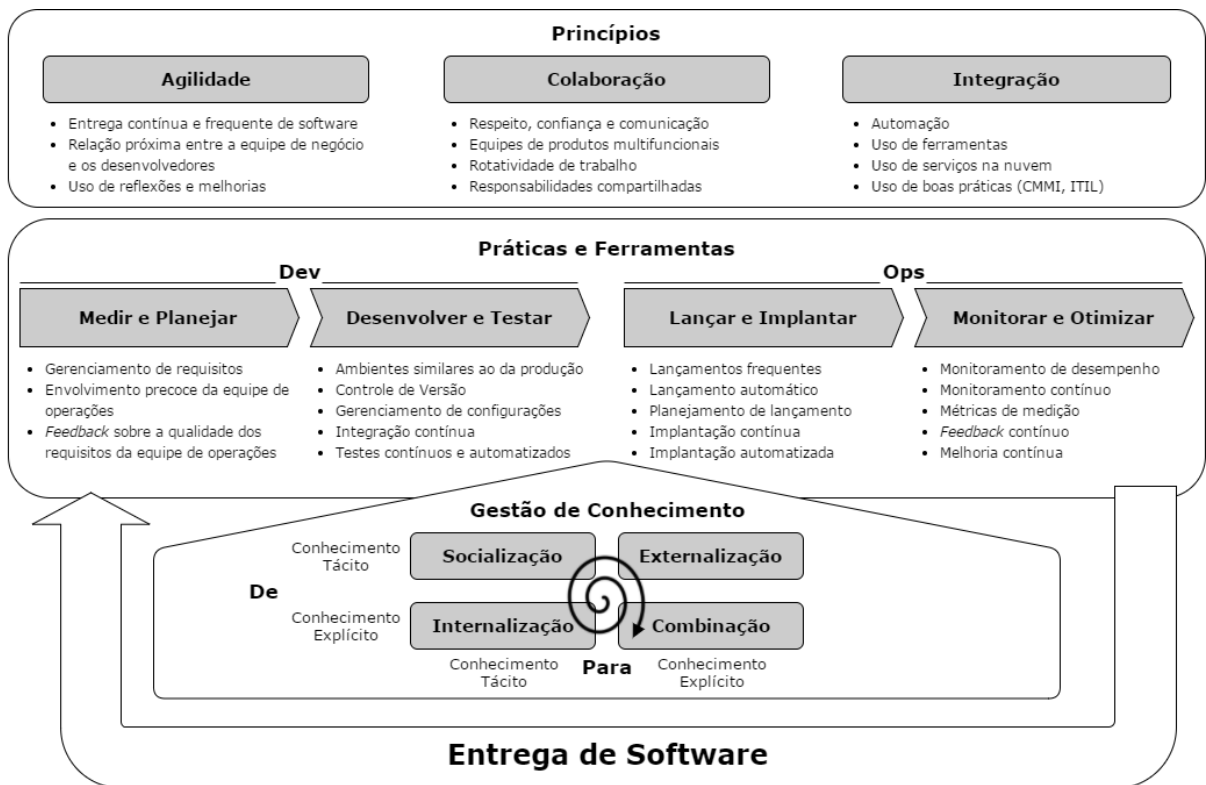


Figura 11: DOKS *Framework* proposto para auxiliar as empresas a evoluir em relação ao DevOps.

Fonte: Adaptado de Nielsen et al. (2017)

Para a avaliação do cumprimento de cada tópico especificado no *framework* são utilizados cinco níveis que determinam o quanto tal atividade ou aspecto é praticado: Não cumprido (*Not fulfilled*), Marginalmente cumprido (*Marginally fulfilled*), Parcialmente cumprido (*Partially fulfilled*), Largamente cumprido (*Largely fulfilled*) e Completamente cumprido (*Fully fulfilled*). A Figura 12 apresenta (a) a legenda da avaliação do DOKS e (b) um exemplo hipotético dessa avaliação.

LEGENDA	
	Não cumprido
	Marginalmente cumprido
	Parcialmente cumprido
	Largamente cumprido
	Completamente cumprido

DESENVOLVER E TESTAR	
Ambientes similares ao da produção	
Controle de versão	
Gerenciamento de configurações	
Integração contínua	
Testes contínuos e automatizados	

(a) Legenda da avaliação do *Framework* DOKS

(b) Avaliação hipotética de tópicos do *Framework* DOKS

Figura 12: Representação da legenda e da avaliação do *Framework* DOKS.

Fonte: Autoria própria

Para a validação do *framework*, os autores do trabalho (NIELSEN et al., 2017) realizaram uma pesquisa e entrevistas com os *stakeholders* de duas empresas, uma de pequeno porte e outra de grande porte. De acordo com o estudo, o *feedback* das empresas foi positivo e o *framework* ajudou a identificar os elementos para o DevOps, bem como elevou o foco para a importância da área de DevOps. Particularmente, a ênfase no compartilhamento de conhecimento mostrou-se útil para explicitar os diferentes modos em que o conhecimento pode ser compartilhado entre as equipes e identificar medidas adequadas de melhoria (NIELSEN et al., 2017).

3 AVALIAÇÕES EMPÍRICAS

Neste capítulo são descritas as avaliações empíricas que foram executadas no decorrer desse trabalho de conclusão de curso. As avaliações em questão foram realizadas com o intuito de alcançar os objetivos propostos no Capítulo 1, tendo como objetivo principal: o fornecimento de um insumo que contribua para a aplicação das práticas e adoção do DevOps utilizando recursos de GC. Na sequência, são explicadas as avaliações empíricas planejadas e executadas no presente estudo.

3.1 PLANEJAMENTO EMPÍRICO

Esta seção apresenta o planejamento das atividades empíricas realizadas. O planejamento é pautado nos objetivos do trabalho e é composto pela execução de um Estudo de Caso em uma empresa real de desenvolvimento de software e a realização de uma *Survey* em âmbito nacional. Complementarmente, em consonância com os objetivos específicos do trabalho, o Quadro 2 relaciona tais objetivos, definidos no Capítulo 1, com as atividades empíricas propostas para atingí-los.

Quadro 2: Relação entre atividades empíricas e objetivos específicos.

Estudo de Caso	<ul style="list-style-type: none"> • Estudar recursos oriundos da área de GC para apoiar a implementação de DevOps, aliviando problemas de adoção da prática; • Aplicar o Framework Conceitual DOKS para avaliar o cumprimento dos principais elementos que compõe a cultura DevOps em um cenário real de desenvolvimento; e • Avaliar a efetividade do framework e o uso de GC para apoiar o DevOps por meio de estudos empíricos e análises qualitativas.
Survey	<ul style="list-style-type: none"> • Mapear as práticas DevOps em casos de sucesso de adoção no cenário nacional; e • Identificar potenciais benefícios e limitações na adoção de DevOps em cenários práticos;

Fonte: Autoria própria

Deste modo, a Seção 3.2 e a Seção 3.3 detalham, respectivamente, as orientações

e procedimentos do Estudo de Caso realizado e o planejamento e execução de uma *Survey* em âmbito nacional, dando ênfase na motivação e nas questões de pesquisa de cada atividade empírica.

3.2 ESTUDO DE CASO

Nessa seção é apresentada a questão de pesquisa, os procedimentos e a estratégia de análise dos dados relacionados ao Estudo de Caso proposto para o trabalho. O estudo de caso pode ser definido como uma metodologia de pesquisa empírica destinada a investigar os fenômenos em um contexto (WOHLIN et al., 2012). O estudo de caso na ES, conforme apontado por Runeson et al. (2012), trata-se de uma investigação empírica baseada em múltiplas fontes de evidência para investigar uma instância de um fenômeno contemporâneo de ES em seu contexto real.

Assim sendo, foi executado um estudo de caso qualitativo em uma empresa real de desenvolvimento de software, utilizando-se de múltiplas fontes de evidências (*Framework* DOKS e questionários) para alcançar os objetivos exibidos no Quadro 2. Uma explicação mais abrangente do Estudo de Caso conduzido é apresentado nas subsequentes Seções.

O *framework* DOKS foi utilizado na íntegra e foi escolhido para o estudo de caso, pois há a carência de estudos e trabalhos que abordam um *framework* conceitual de GC para apoiar as práticas DevOps e pelo DOKS ser específico para auxiliar na evolução e adoção do DevOps em organizações com foco na GC.

3.2.1 DESIGN DA PESQUISA

A seguir é explicado de forma sumarizada o projeto experimental idealizado, expondo a questão de pesquisa e o objeto de estudo, entretanto mais detalhes e informações sobre essa atividade empírica estão no Protocolo de Estudo de Caso (Apêndice A).

O planejamento e execução do estudo de caso é motivado pela seguinte questão de pesquisa: O *framework* conceitual DOKS, proposto para a avaliação e viabilização da adoção da cultura DevOps em organizações desenvolvedoras de software com foco na GC, é eficiente e eficaz ao seu propósito?

O objeto de estudo dessa avaliação empírica trata-se de uma empresa desenvolvedora de software de grande porte, que fornece soluções para supermercados, restaurantes e materiais de construção. A empresa está situada no Sudoeste do estado do Paraná, mas tem atuação em

todos os estados do Brasil, contando com mais de 500 funcionários, mais de 5 mil clientes e mais de 100 mil usuários. Por questões de sigilo, no decorrer deste documento a empresa será chamada de Empresa A.

O estudo de caso foi realizado no departamento de Novos Produtos, o qual é responsável pelo desenvolvimento das novas soluções oferecidas pela Empresa A. Para a pesquisa participaram três equipes que desenvolvem um novo produto em específico, essas três equipes representam um total de 25 funcionários. Adicionalmente, tais equipes já trabalham com metodologias ágeis e estão iniciando o trabalho com as práticas da cultura DevOps.

3.2.2 EXECUÇÃO DO ESTUDO DE CASO

A execução do estudo de caso teve início no dia 20 de Agosto e finalizou no dia 13 de Novembro, sendo dividida em duas etapas: (i) levantar dados iniciais das equipes acerca do DevOps e GC, por meio de um questionário. Posteriormente, aplicar o *framework* DOKS com alguns integrantes das equipes e realizar a avaliação das diretrizes e princípios que o trabalho de Nielsen et al. (2017) expõe no *framework*. Por fim, analisar os dados obtidos e propor melhorias conforme as lacunas encontradas; e (ii) após a realização das melhorias, reavaliar as equipes acerca das melhorias propostas, bem como reaplicar o DOKS para validar se houve evolução em algumas diretrizes ou princípios. A Figura 13 ilustra e enumera o fluxo citado em macro atividades.

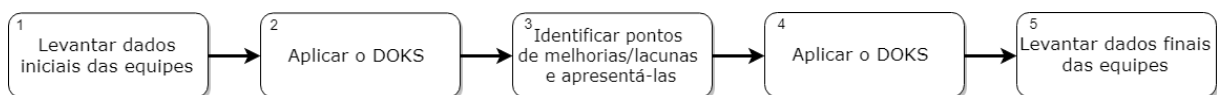


Figura 13: Fluxo das atividades realizadas no Estudo de Caso.

Fonte: Autoria própria

Para levantar dados iniciais das equipes (atividade 1) e obter uma visão geral das equipes acerca do DevOps e da GC, foi utilizado um questionário, que é composto por perguntas objetivas e subjetivas. As perguntas que fizeram parte do questionário seguiram os seguintes tópicos: (1) levantamento do perfil do profissional; (2) histórico e experiência com o DevOps; (3) visão das práticas, ferramentas ou tecnologias executadas no ambiente e equipe de trabalho; e (4) como é percebido e praticado a GC nas equipes. O Questionário 1 foi disponibilizado para todos os integrantes das equipes e é apresentado de forma completa no Protocolo de Estudo de Caso (Apêndice A).

Em relação à aplicação do *framework* DOKS (atividade 2), ela foi realizada de forma

interna com pessoas-chave das equipes. Essa decisão foi tomada pois no trabalho original do *framework* (NIELSEN et al., 2017) a avaliação das diretrizes e princípios é realizada pelos autores de maneira externa à empresa. Entretanto, os autores sugerem que essa avaliação possa ser feita de maneira interna, desde que envolva os funcionários da empresa, da equipe de Dev e da equipe de Ops. Na primeira aplicação do *framework*, o intuito era avaliar as diretrizes, princípios e aspectos relacionados ao DevOps elencados no *framework* e que estão presentes nas equipes

A etapa de identificar pontos de melhorias/lacunas e apresentá-las (atividade 3) foi composta pela análise dos dados do primeiro questionário e da primeira aplicação do DOKS. Com essa análise foi possível detectar algumas lacunas que estavam presentes nas equipes, adicionalmente, na apresentação dessas lacunas identificadas para a gerência foi apresentado medidas que poderiam ser tomadas para auxiliar na correção e reparação dessas lacunas.

Posteriormente, foi novamente aplicado o *framework* DOKS (atividade 4) seguindo os mesmos moldes da primeira aplicação. O objetivo da segunda aplicação era avaliar a evolução de princípios e diretrizes após a proposição e realização de melhorias.

Para levantar dados finais das equipes (atividade 5) e avaliar as melhorias sugeridas foi utilizado um segundo questionário, que também é composto por perguntas objetivas e subjetivas. A estrutura do questionário e das perguntas foram: (1) identificação do profissional; (2) opinião sobre ações para melhorar o quesito de “Melhoria contínua e reflexão”; (3) opinião sobre ações para melhorar o quesito da “Cultura DevOps”; (4) opinião sobre ações para melhorar o quesito de GC; e (5) opinião sobre ações para melhorar o quesito de “Monitorar e Otimizar”. O Questionário 2 foi disponibilizado para todos os integrantes das equipes e é apresentado de forma completa no Protocolo de Estudo de Caso (Apêndice A).

Por decisão de projeto, optou-se por não relacionar os tópicos do primeiro questionário com o os tópicos do segundo questionário, haja vista que o primeiro questionário tem o foco de reconhecimento do perfil do profissional, levantando sua experiência com DevOps e a prática de GC nas equipes, enquanto o segundo apenas identifica o profissional e solicita sua opinião sobre as melhorias que foram propostas. Desse modo, avaliar a relação da experiência do profissional com a opinião dele sobre os tópicos não era o objetivo da pesquisa.

3.2.3 ESTRATÉGIA PARA ANÁLISE DOS DADOS

A análise dos dados do Estudo de Caso foi realizada de forma qualitativa e quantitativa, sendo os dados sumarizados em gráficos e quadros para melhor visualização e entendimento. O

método utilizado para a análise dos dados é o de Triangulação, que consiste em reunir diferentes tipos de evidências para apoiar uma proposição (SEAMAN, 1999).

A Triangulação permite confirmar a validade das conclusões, pois se a mesma conclusão pode ser tirada de várias fontes de informação, ou seja, passando pelo processo de Triangulação, essa conclusão é mais forte do que uma conclusão baseada em uma única fonte (SEAMAN, 1999; WOHLIN et al., 2012). Em um estudo de caso, é importante levar em conta pontos de vista de papéis diferentes e investigar diferenças, pois comumente, as conclusões são tiradas analisando as diferenças entre as fontes de dados (WOHLIN et al., 2012).

Na pesquisa atual, a análise dos resultados obtidos por meio da execução do estudo de caso tem por finalidade responder a questão de pesquisa proposta para essa atividade empírica. Desse modo, os dados obtidos na primeira parte e na segunda parte do estudo, serão comparados para verificar se houve melhorias. Adicionalmente, informações e feedback obtidos da Empresa A serão contemplados na análise, para possibilitar uma validação dos resultados obtidos.

3.3 PLANEJAMENTO E EXECUÇÃO DA *SURVEY*

Nessa seção é apresentada as questões de pesquisa, a estrutura, a estratégia de coleta de dados e a estratégia de análise de respostas relacionados à *Survey*. Uma *survey* pode ser definida como um meio de coleta de informações de determinado grupo de pessoas sobre suas ações, comportamentos ou opiniões por meio de um instrumento de pesquisa, geralmente questionários (FREITAS et al., 2000; WOHLIN et al., 2012). Em relação ao seu propósito, a *survey* pode ser classificada em descritiva, explanatória e exploratória (WOHLIN et al., 2012).

As *surveys* descritivas são conduzidas para identificar as situações, eventos, atitudes ou opiniões manifestados em uma população (FREITAS et al., 2000) e possibilitar realizar afirmações sobre as populações de estudo (WOHLIN et al., 2012). As *surveys* explanatórias tem como objetivo testar uma teoria e as relações causais, bem como questionar o porquê da existência dessa relação (FREITAS et al., 2000). As *surveys* exploratórias são usados como pré-estudo para uma investigação mais completa (WOHLIN et al., 2012), o objetivo é familiarizar-se com o tópico de estudo ou identificar conceitos iniciais sobre um tópico (FREITAS et al., 2000).

Portanto, com a intenção de atingir os objetivos específicos definidos no Quadro 2, foi conduzida uma *Survey* do tipo descritiva em âmbito nacional. A explicação mais abrangente do planejamento e execução da *Survey* são apresentas nas subseqüentes seções.

3.3.1 QUESTÕES DE PESQUISA

O planejamento e execução da *Survey* são motivados pelas seguintes questões de pesquisa:

- **QP1:** Qual o cenário da cultura DevOps em um nível nacional?
- **QP2:** Quais são as ferramentas mais utilizadas nas principais práticas da cultura DevOps?
- **QP3:** Qual o cenário da GC em um nível nacional?
- **QP4:** Quais são as principais práticas de GC das organizações?
- **QP5:** Quais são os benefícios na adoção da cultura DevOps?
- **QP6:** Quais são as limitações que impedem a adoção efetiva da cultura do DevOps?

3.3.2 ESTRUTURAÇÃO DA *SURVEY*

A *Survey* foi elaborada com auxílio de uma ferramenta online para a criação de formulários¹ e sua estrutura está de acordo com a especificação completa dos tópicos e das perguntas representados no Apêndice B.

Os tópicos definidos para a *Survey* são: (Tópico 1) levantamento do perfil do profissional e da empresa que trabalha; (Tópico 2) conhecimento e experiência sobre o movimento cultural DevOps; (Tópico 3) visão da cultura DevOps adotada e realizada na empresa; (Tópico 4) práticas e ferramentas utilizadas na empresa que dão suporte e estão presentes na cultura DevOps; (Tópico 5) visão sobre a GC praticada na empresa; (Tópico 6) benefícios notados com a adoção da cultura DevOps; e (Tópico 7) limitações e desafios para a adoção da cultura.

No Quadro 3 é feito o relacionamento entre os tópicos e as questões de pesquisa que eles visam responder, exceto o Tópico 1, que é voltado para questões demográficas.

¹ Kiwik Surveys - <https://kwiksurveys.com/>

Quadro 3: Relacionamento entre os tópicos e questões de pesquisa da *Survey*.

Tópicos	Questões de Pesquisa
Tópico 1	Questão demográfica
Tópico 2	QP1
Tópico 3	QP1
Tópico 4	QP2
Tópico 5	QP3 e QP4
Tópico 6	QP5
Tópico 7	QP6

Fonte: Autoria própria

3.3.3 ESTRATÉGIAS DE COLETA E ANÁLISE DOS DADOS

A coleta de respostas para a *Survey* iniciou com sua disponibilização e distribuição online no dia 23 de Setembro até o dia 05 de Novembro. As *surveys* online são vantajosas para o respondente, por facilitar o acesso e utilização, e para o pesquisador, por disponibilizar os dados de forma eletrônica (PUNTER et al., 2003). Assim sendo, a *Survey* foi disponibilizada por meio de *e-mails* e redes sociais profissionais para profissionais que trabalham na área de TI.

A *Survey* foi executada para acumular uma quantidade satisfatória de respostas, a fim de possibilitar uma análise quantitativa e qualitativa dos dados com qualidade e que forneça insumos relevantes para alcançar os objetivos estipulados para esse método de pesquisa. A estratégia de análise das respostas da *Survey* está pautada em sumarizar os dados obtidos em gráficos e quadros, para facilitar a visualização e compreensão dos mesmos.

4 RESULTADOS E DISCUSSÕES

Este capítulo apresenta os resultados obtidos a partir da execução das avaliações empíricas descritas no Capítulo 3 e realiza análise dos dados e a discussão dos resultados obtidos. Em particular, a Seção 4.1 apresenta os resultados do Estudo de Caso, a Seção 4.2 apresenta os resultados da *Survey* de nível nacional, a Seção 4.3 apresenta as respostas das questões de pesquisa e discussões adicionais, a Seção 4.4 apresenta as ameaças à validade do trabalho e, por fim, a Seção 4.5 apresenta as publicações realizadas a partir dos resultados obtidos com a pesquisa.

4.1 RESULTADOS DO ESTUDO DE CASO

Esta seção possui os resultados obtidos no Estudo de Caso. Para facilitar a visualização, os dados foram agrupados conforme o meio para a coleta dos dados, dividindo-se em Questionário 1, Questionário 2 e as avaliações com o *Framework* DOKS. Entretanto, antes da apresentação dos resultados será apresentado as melhorias propostas conforme as lacunas encontradas a partir da análise dos resultados obtidos.

4.1.1 LACUNAS IDENTIFICADAS E MELHORIAS PROPOSTAS

As lacunas foram identificadas após a análise dos dados obtidos no Questionário 1 e da primeira avaliação do *framework* DOKS, conforme exposto na Figura 13. A seguir estão elencadas as lacunas identificadas:

- **Lacuna 1 - Melhoria contínua e reflexões:** esta lacuna foi identificada a partir da avaliação do DOKS e do Questionário 1, no qual este aspecto teve baixa avaliação em ambas as avaliações empíricas;
- **Lacuna 2 - Aspecto Cultural do DevOps:** esta lacuna foi identificada a partir da avaliação do DOKS e do Questionário 1, no qual o problema identificado não estava

ligado à ferramentas ou práticas com as ferramentas e, sim, ligado ao conhecimento e consciência dos aspectos da cultura DevOps;

- **Lacuna 3 - Melhoria e incentivo à GC:** esta lacuna foi identificada principalmente pelas respostas do Questionário 1, no qual os colaboradores apontam um déficit na GC da Empresa A; e
- **Lacuna 4 - Monitorar e otimizar:** esta lacuna foi identificada a partir da avaliação do DOKS e do Questionário 1, no qual foi visto que as equipes não praticam de forma efetiva a ação de monitorar e otimizar o sistema.

A partir das lacunas, foram realizadas sugestões de melhorias para auxiliar a sanar ou melhorar os problemas encontradas. o Quadro 4 apresenta a relação entre as lacunas e as melhorias propostas, bem como a justificativa para tal sugestão de melhoria.

Quadro 4: Relação entre as lacunas e as sugestões de melhorias.

Lacunas	Melhorias propostas
Lacuna 1	<p>Ações propostas para melhoria contínua e reflexões nas equipes: (i) <i>Grooming</i> mais técnico; (ii) Reunião para análise da situação da Sprint no início da segunda semana; (iii) Respeitar o time-boxed das cerimônias estipulados pelo guia do Scrum; (iv) Análise das planilhas de retrospectiva das Sprints anteriores; e (v) Caixa de sugestões.</p> <p>As propostas de melhorias foram elencadas a partir da sugestão de um especialista da área de processos da própria Empresa A. Tais ações não são mandatórias e ficou aberto para as equipes decidirem as melhores abordagens para fomentar a melhoria contínua e reflexões, tendo em vista que esse problema foi identificado.</p>
Lacuna 2	<p>Ações propostas para melhorar o aspecto cultural do DevOps: (i) Apresentação sobre o DevOps para as equipes (o que é, quem utiliza, benefícios); e (ii) Artigo no blog detalhando o DevOps (o que é, fundamentos e práticas).</p> <p>Tais ações foram sugeridas, pois esse artifício de apresentação já é utilizado pela Empresa A para compartilhar conhecimento e o blog é uma ferramenta que começou a ser utilizada pela Empresa A.</p>
Lacuna 3	<p>Ações propostas para melhorar e incentivar a GC: (i) Reforçar o uso do blog; (ii) Incentivar o pessoal a participar publicando e lendo no blog; e (iii) Escrever no blog sobre a arquitetura do sistema.</p> <p>Tais ações foram sugeridas, pois as equipes já estavam planejando e iniciando a utilização de um blog.</p>
Lacuna 4	<p>Ações propostas para monitorar e otimizar o sistema: (i) Expor dados do sistema de maneira dinamicamente por meio de monitores ou TV.</p> <p>Tal ação foi baseada conforme sugestão dos autores Davis e Daniels (2016).</p>

Fonte: Autoria própria

4.1.2 QUESTIONÁRIO 1

Para realizar a avaliação e obter uma visão geral da Cultura DevOps e da GC nas equipes que desenvolvem um produto específico para a Empresa A, foi disponibilizado para os funcionários dessas equipes um questionário. Este não tinha teor de obrigatoriedade e obteve um total de 18 respostas.

Os dados acerca da experiência profissional do funcionário estão expostos na Figura 14, na qual são apresentadas as respostas obtidas para as perguntas referentes ao (a) tempo como profissional de TI e ao (b) tempo de trabalho exercido para a Empresa A.

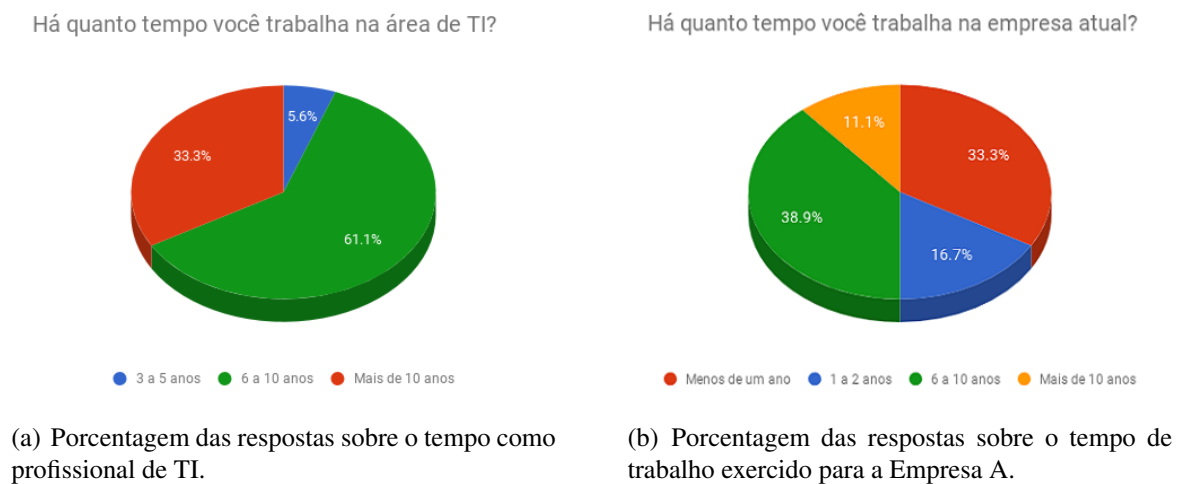


Figura 14: Gráficos sobre a experiência dos funcionários na área de TI e na Empresa A.

Fonte: Autoria própria

Em relação ao perfil dos profissionais das três equipes, nota-se que dos 18 profissionais, 94,4% possuem mais de seis anos trabalhando na área de TI e desses profissionais, apenas 33,3% trabalham há menos de um ano na Empresa A. Portanto, são profissionais que possuem conhecimento e experiência técnica e estão habituados aos processos e diretrizes da empresa.

Associado ao tópico sobre histórico e experiência com DevOps, a Figura 15 expõe as respostas relativas ao fato do funcionário já ter ouvido falar do termo DevOps, no qual as respostas “Não” e “Talvez” atingiram o mesmo percentual (5,6%) e o Quadro 5 representa as respostas obtidas ao instigar o funcionário a explicar e definir o significado do DevOps, no total obteve-se 8 respostas.

Diante disso, percebe-se que dentre os 18 funcionários, aproximadamente 16 (88,9%) já ouviram falar do termo DevOps. Contudo, apenas 8 dentre os 18 participantes desse primeiro questionário tentaram definir o termo DevOps, apesar de já estarem inseridos em um ambiente

DevOps em estágio inicial.

Você já ouviu falar do termo DevOps?

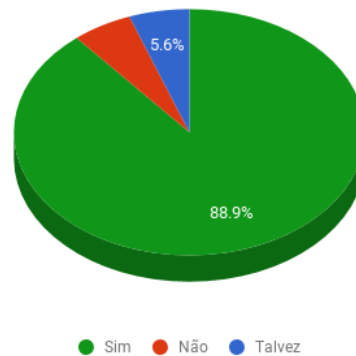


Figura 15: Gráfico sobre a questão de ter ouvido falar do termo DevOps.

Fonte: Autoria própria

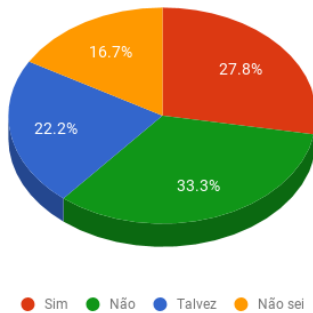
Quadro 5: Definição do termo DevOps pelo funcionário.

“ <i>Framework</i> de Integração entre as áreas de Desenvolvimento e Operação.”
“Entrega contínua.”
“Operações de suporte ao ambiente de desenvolvimento.”
“Conjunto de processos e ferramentas que garantam uma sincronia harmoniosa entre desenvolvimento e operação visando alcançar melhores resultados de execução de trabalho e entrega de valor.”
“Basicamente, seria uma metodologia/prática utilizada no desenvolvimento de software para que a equipe de desenvolvimento e infraestrutura/ <i>deploy</i> consigam ter uma maior integração, trabalhando em conjunto e de forma mais sincronizada, com o objetivo de diminuir o tempo das entregas para produção.”
“Capacidade de produzir e gerenciar o desenvolvimento de software em todas as suas fases de forma automática e sincronizada, cobrindo todos os processos.”
“Em poucas palavras, um conjunto de ferramentas para auxiliar/automatizar processos de desenvolvimento de software.”
“Uma Cultura Organizacional.”

Fonte: Autoria própria

Ainda relacionado ao tópico sobre histórico e experiência com DevOps, é visto na Figura 16 as respostas pertinente à: (a) experiência do funcionário com práticas da Cultura DevOps e (b) se a equipe/produto que atualmente ele trabalha utiliza práticas da cultura DevOps.

Você já trabalhou com práticas da cultura DevOps?



(a) Porcentagem dos participantes que consideram terem trabalhado com a Cultura DevOps.

Na sua opinião, a equipe/produto que atualmente você trabalha utiliza práticas da cultura DevOps?



(b) Porcentagem das respostas acerca da visão do DevOps na equipe/produto.

Figura 16: Gráficos sobre a experiência dos funcionários com o DevOps.

Fonte: Autoria própria

Os dados apresentados indicam que apesar dos profissionais experientes e 88,9% terem ouvido falar do termo, um total de 27,7% afirma com certeza ter trabalhado com práticas da cultura DevOps. Adicionalmente, dos 18 funcionários, um total de 38,9% acreditam com certeza que sua equipe ou as equipes que desenvolvem o produto utilizam práticas da cultura DevOps.

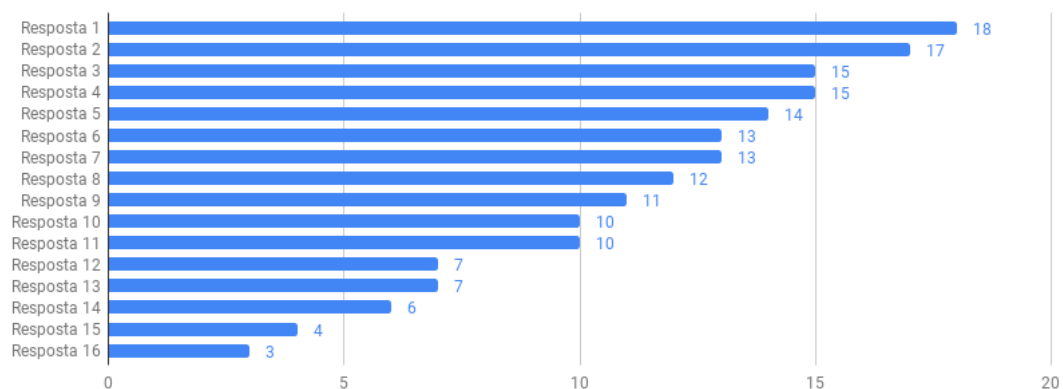
A resposta sobre o tópico da visão dos funcionários no tocante às práticas, ferramentas ou tecnologias praticadas, executadas ou usadas no ambiente e equipe de trabalho é apresentada na Figura 17. A legenda das respostas foram dispostas no Quadro 6, acompanhando a Figura 17, para melhorar a visualização dos dados.

Quadro 6: Legenda da pergunta sobre práticas, ferramentas ou tecnologias adicionais do DevOps.

Legenda	Resposta Original
Resposta 1	Controle de versão
Resposta 2	Uso de serviços na nuvem
Resposta 3	Comunicação entre as equipes
Resposta 4	Gestão de requisitos
Resposta 5	Gerenciamento de configurações
Resposta 6	Comunicação entre indivíduos
Resposta 7	União da equipe de desenvolvimento e equipe de operação
Resposta 8	Ambientes similares ao da produção
Resposta 9	Testes contínuos e automatizados
Resposta 10	Compartilhamento do conhecimento entre indivíduos e equipes
Resposta 11	Integração contínua
Resposta 12	Entrega contínua
Resposta 13	Implantação automatizada
Resposta 14	Implantação contínua
Resposta 15	Monitoramento do desempenho do sistema
Resposta 16	Melhoria e reflexões contínua

Fonte: Autoria própria

Na sua opinião, quais dessas práticas, ferramentas ou tecnologias são encontradas no ambiente e equipe de trabalho?

**Figura 17: Gráfico sobre as práticas, ferramentas e tecnologias do DevOps na Empresa.**

Fonte: Autoria própria

Sobre o tópico concernente à GC nas equipes, a Figura 18 exibe a opinião dos funcionários sobre a realização da GC na(s) equipe(s) nas quais eles trabalham. Nessa questão, a opção “Não sei opinar sobre o assunto” obteve 5,6% das respostas.

Na sua opinião, há a gestão do conhecimento sobre processos, negócios, tecnologias e ferramentas na(s) equipe(s) que você atualmente trabalha?



Figura 18: Gráfico sobre a existência da GC nas equipes.

Fonte: Autoria própria

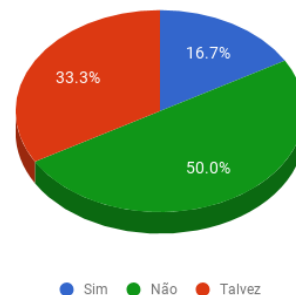
A Figura 19 apresenta as respostas sobre as questões de (a) existir práticas ou ferramentas para auxiliar a GC e (b) se as ferramentas são utilizadas de forma adequada e frequente. Por fim, o Quadro 7 apresenta um comentário adicional em relação à GC.

A empresa/equipe utiliza práticas e ferramentas que auxiliam na gestão do conhecimento?



(a) Porcentagem das respostas sobre o uso de práticas e ferramentas da GC na Empresa A.

Na sua opinião, tais ferramentas são utilizadas adequadamente e frequentemente?



(b) Porcentagem das respostas sobre o uso adequado das ferramentas de GC.

Figura 19: Gráficos sobre as ferramentas e práticas da GC na Empresa A.

Fonte: Autoria própria

Quadro 7: Comentário adicional sobre a GC.

“Falta difundir conhecimento sobre arquitetura do live, temos pouca documentação de exemplos de código.”

Fonte: Autoria própria

Diante das respostas referentes à GC, pode-se visualizar que apesar de existir algumas práticas e ferramentas que fomentam aspectos da GC, metade dos funcionários acreditam que não está sendo realizada e utilizada de maneira adequada e frequente as práticas e ferramentas, bem como, 7 (38,9%) dos 18 funcionários acreditam que há a GC relativo a processos, regras de negócios, tecnologias e ferramentas. Nesse caso, é notável um déficit em relação à GC aplicada nas equipes.

4.1.3 QUESTIONÁRIO 2

Para realizar a avaliação das melhorias propostas foi disponibilizado para os funcionários dessas equipes um segundo questionário. Este não tinha teor de obrigatoriedade e obteve um total de 14 respostas. Entretanto, vale ressaltar que, durante o período de execução do estudo dois funcionários deixaram a empresa e outros dois funcionários foram migrados para uma equipe que desenvolve outro produto da Empresa A.

De maneira geral, as questões aplicadas neste questionário foram voltadas para o quanto o funcionário concordava com determinada afirmação numa escala de 1 (nível mais baixo) a 5 (nível mais alto) e questões abertas para o funcionário expor sua opinião, crítica ou sugestão sobre o tópico a ser respondido e sobre a pesquisa. A seguir, são apresentadas as respostas do segundo questionário.

Referente às ações tomadas para a melhoria contínua, a Figura 20 apresenta a avaliação dos funcionários para a afirmação “As ações realizadas tiveram um efeito positivo para a melhoria contínua e reflexões nas equipes” e o Quadro 8 apresenta comentários adicionais sobre as ações.

Em uma escala de 1 a 5, o quanto você concorda com a seguinte afirmação: "As ações realizadas tiveram um efeito positivo para a melhoria contínua e reflexões nas equipes."

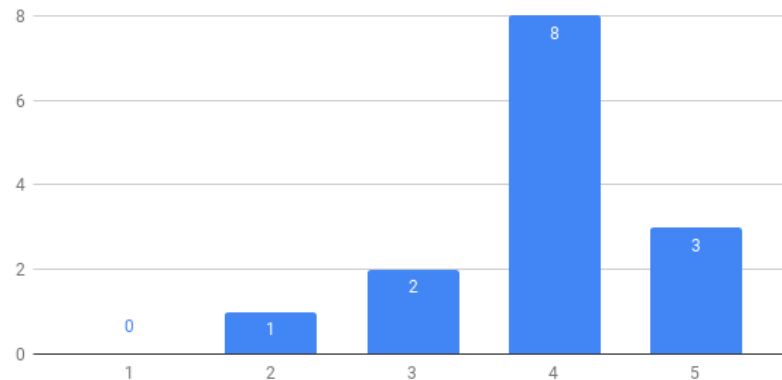


Figura 20: Gráfico das respostas sobre ações de Melhoria contínua e reflexões.

Fonte: Autoria própria

Quadro 8: Comentários adicionais sobre a Melhoria contínua e reflexões.

“Não observei a ação: - Reunião para análise da situação da Sprint no início da segunda semana; em nossa equipe.”
“Um ganho que foi legal é a questão de cada membro do time ser responsável por uma atividade, talvez possa ser usado.”

Fonte: Autoria própria

De acordo com o resultado, pode-se averiguar que as ações realizadas para a melhoria contínua e reflexões tiveram um efeito positivo nesse aspecto. Em relação aos comentários, deixou-se aberto para as equipes e líderes de equipes tomarem iniciativas que contribuíssem para a melhoria contínua e reflexões. Por esse motivo, algumas ações podem ter sido realizadas em algumas equipes e em outras não.

Relativo as ações tomadas para aprimorar o aspecto cultural do DevOps nas equipes, a Figura 21 apresenta a avaliação dos funcionários para a afirmação “A equipe/produto que atualmente você trabalha teve melhorias na cultura DevOps”. Vale destacar que a pergunta para sugestões e comentários não houve resposta. Conforme o resultado apresentado, as ações tomadas para o aspecto cultural do DevOps tiveram efeitos positivos para a melhoria desse ponto.

Em uma escala de 1 a 5, o quanto você concorda com a seguinte afirmação: "A equipe/produto que atualmente você trabalha teve melhorias na cultura DevOps."

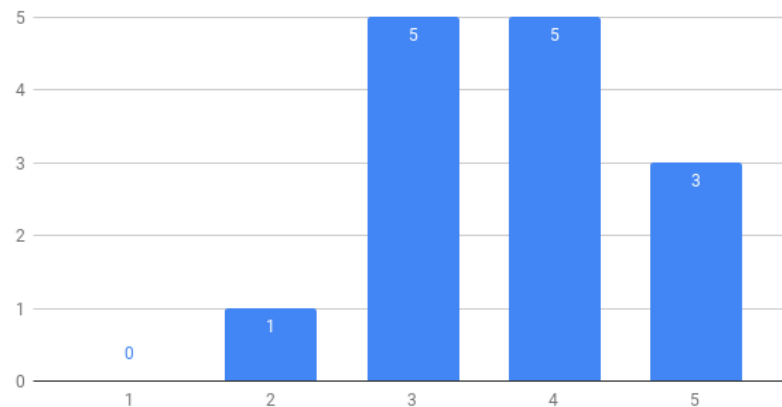
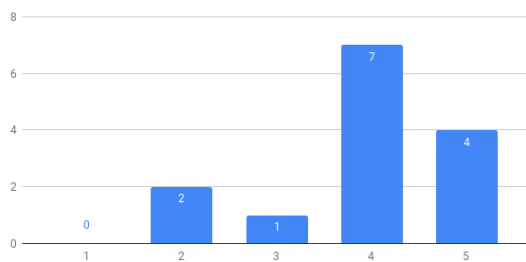


Figura 21: Gráfico das respostas sobre ações para a cultura DevOps.

Fonte: Autoria própria

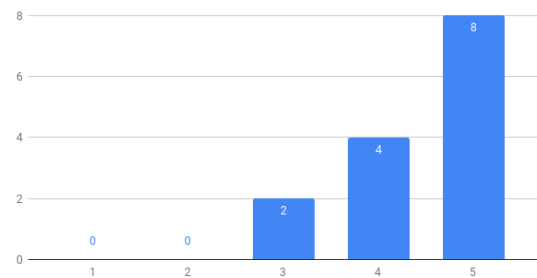
Em relação às ações tomadas para aprimorar a GC, a Figura 22 apresenta: (a) a avaliação dos funcionários para a afirmação “As ações realizadas tiveram um efeito positivo para melhorar a Gestão de Conhecimento”; e (b) a avaliação para a afirmação “O blog é uma ferramenta útil e boa para compartilhar o conhecimento”. O Quadro 9 apresenta os comentários adicionais sobre o blog e a GC.

Em uma escala de 1 a 5, o quanto você concorda com a seguinte afirmação: "As ações realizadas tiveram um efeito positivo para melhorar a Gestão de Conhecimento."



(a) Respostas obtidas sobre as ações para a GC.

Em uma escala de 1 a 5, o quanto você concorda com a seguinte afirmação: "O blog é uma ferramenta útil e boa para compartilhar o conhecimento."



(b) Respostas obtidas sobre o blog.

Figura 22: Gráficos sobre as ações para GC e blog.

Fonte: Autoria própria

Quadro 9: Comentários adicionais sobre o blog e GC.

“Concordo com a afirmação de que o blog é uma excelente forma de adquirir conhecimento, porém as pessoas com o conhecimento tácito precisam ter/gastar um tempo para repassá-lo para explícito, e muita vezes esse tempo não temos dentro da sprint. Sugiro poder trabalhar no blog de casa, fazendo esse conteúdo em casa e essas horas vir para banco ou hora extra.”

“Algumas rotinas mais complexas deveriam estar no blog também, não apenas postagens superficiais de desenvolvimento.”

Fonte: Autoria própria

As ações tomadas para o incentivo da GC dentro das equipes também obtiveram efeitos positivos. Os funcionários aprovaram o uso do blog e foi fundamental para a melhoria da GC, entretanto, por se tratar de uma ferramenta interna os funcionários não tem acesso externo, limitando a colaboração dos mesmos com postagens no blog e um outro comentário sugere que sejam feitas postagens mais complexas e não apenas superficiais.

Referente às ações para melhorar o quesito de “Monitorar e Otimizar” é importante destacar que nenhuma ação foi de fato realizada e as perguntas se baseiam de forma hipotética sobre os possíveis resultados das ações propostas. Dessa forma, a Figura 23 apresenta a avaliação para a afirmação “A ação proposta pode ter um efeito positivo para melhorar o monitoramento e otimização do sistema” e o Quadro 10 apresenta os comentários adicionais sobre o monitoramento e otimização do sistema.

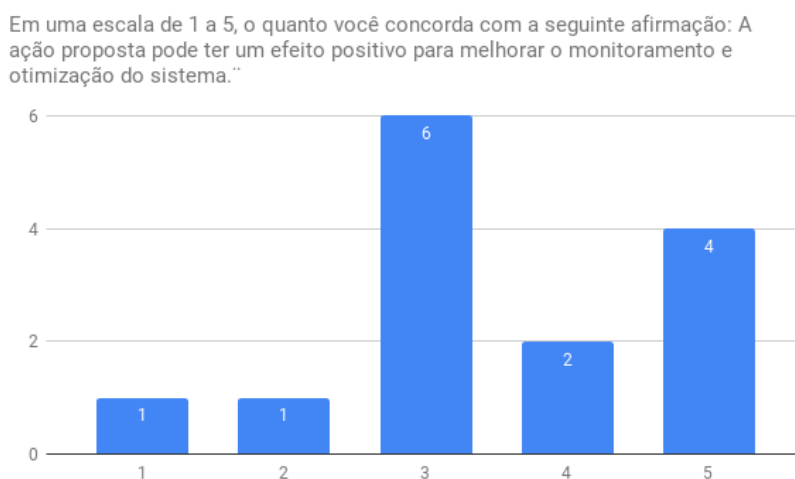


Figura 23: Gráfico das respostas sobre monitoramento e otimização do sistema.

Fonte: Autoria própria

Quadro 10: Comentários adicionais sobre o monitoramento e otimização do sistema.

“Antes de ter dados para serem expostos e gerenciado, devemos primeiramente ser mais organizados, ou seja, de vez gastar esforços para apresentar dados para monitoramento, deve aprender a coletar os dados para somente após isso apresentar.”
“Não foi feito nenhuma ação de monitoramento.”

Fonte: Autoria própria

O monitoramento e otimização do sistema continua sendo um déficit, pois as ações para melhorar esse aspecto demandavam mais custos e tempo, o que dificulta a sua execução imediata. Portanto, é um ponto a ser melhorado e merece atenção diante dos comentários feitos pelos funcionários.

Por fim, havia uma questão aberta, dando espaço para o funcionário deixar sua opinião, crítica ou sugestão referente à pesquisa em si. O Quadro 11 apresenta as respostas deixadas pelos funcionários. Tais comentários indicam que a pesquisa foi importante e sugere um ponto que poderia ter sido executado durante a pesquisa, porém não houve o planejamento para tal medida.

Quadro 11: Comentários adicionais sobre a pesquisa.

“De maneira geral, esse trabalho que vem sendo realizado está muito legal e bacana, passa uma sensação firme de preocupação com o nosso dia-a-dia, tecnologia, processo e etc. Parabéns aos envolvidos.”
“Acredito que as ações deveriam ser discutidas nas reuniões de abertura/encerramento das sprints.”

Fonte: Autoria própria

4.1.4 *FRAMEWORK DOKS*

As respostas obtidas por meio da aplicação do *Framework DOKS* são apresentadas nesta Seção. A apresentação de tais resultados será dividido conforme as seções que o DOKS possui: (i) Princípios, (ii) Práticas e Ferramentas e (iii) GC. A representação gráfica dos resultados seguirá uma representação similar ao trabalho original (NIELSEN et al., 2017) e a legenda dos resultados está disposta na Figura 12, que representa o nível que aquele tópico é cumprido dentro das equipes.

4.1.4.1 PRINCÍPIOS

Em relação ao Princípios, eles eram divididos em Agilidade, Colaboração e Automação. A seguir, a Figura 24 apresenta os resultados para cada divisão dos Princípios

no (a) início e (b) final do estudo. Os resultados apresentados demonstram que as avaliações dos tópicos para a Agilidade, Colaboração e Integração se mantiveram estáveis, o que pode ser considerado como um ponto positivo, existindo apenas uma melhora no quesito “Respeito, confiança e comunicação”.

AGILIDADE		COLABORAÇÃO		INTEGRAÇÃO	
Entrega contínua e frequente de software		Respeito, confiança e comunicação		Automação	
Relação próxima entre a equipe de negócio e os desenvolvedores		Equipes multifuncionais		Uso de ferramentas	
Uso de reflexões e melhorias		Rotatividade de trabalho		Uso de serviços na nuvem	
		Responsabilidades compartilhadas		Uso de boas práticas (CMMI, ITIL)	

(a) Avaliação dos Princípios no início do estudo.

AGILIDADE		COLABORAÇÃO		INTEGRAÇÃO	
Entrega contínua e frequente de software		Respeito, confiança e comunicação		Automação	
Relação próxima entre a equipe de negócio e os desenvolvedores		Equipes multifuncionais		Uso de ferramentas	
Uso de reflexões e melhorias		Rotatividade de trabalho		Uso de serviços na nuvem	
		Responsabilidades compartilhadas		Uso de boas práticas (CMMI, ITIL)	

(b) Avaliação dos Princípios no final do estudo.

Figura 24: Avaliação dos Princípios do *Framework* DOKS.

Fonte: Autoria própria

4.1.4.2 PRÁTICAS E FERRAMENTAS

Em relação às Práticas e Ferramentas, esta seção no *framework* está subdividida entre as práticas e ferramentas do Dev e as do Ops. Portanto, a Figura 25 apresenta a avaliação das práticas e ferramentas do Dev, mostrando a avaliação no (a) início e (b) final do estudo. Já a Figura 26 apresenta a avaliação das práticas e ferramentas do Ops, mostrando a avaliação no (a) início e (b) final do estudo.

Referente às práticas e ferramentas da equipe de Dev (Figura 25), houve a melhoria no tópico “Feedback sobre a qualidade dos requisitos da equipe de operações”, que compreende a participação da equipe de Operações na avaliação e comentários dos requisitos propostos para o sistema. Referente às práticas e ferramentas da equipe de Ops (Figura 26), houve a melhoria nos tópicos “Implantação contínua”, “Feedback contínuo” e “Melhoria Contínua”. Os tópicos restantes se mantiveram estáveis, o que é considerado positivo.

MEDIR E PLANEJAR		DESENVOLVER E TESTAR	
Gerenciamento de requisitos		Ambientes similares ao da produção	
Envolvimento precoce da equipe de operações		Controle de versão	
Feedback sobre a qualidade dos requisitos da equipe de operações		Gerenciamento de configurações	
		Integração contínua	
		Testes contínuos e automatizados	

(a) Avaliação das práticas e ferramentas do Dev no início do estudo.

MEDIR E PLANEJAR		DESENVOLVER E TESTAR	
Gerenciamento de requisitos		Ambientes similares ao da produção	
Envolvimento precoce da equipe de operações		Controle de versão	
Feedback sobre a qualidade dos requisitos da equipe de operações		Gerenciamento de configurações	
		Integração contínua	
		Testes contínuos e automatizados	

(b) Avaliação das práticas e ferramentas do Dev no final do estudo.

Figura 25: Avaliação as práticas e ferramentas do Dev no *Framework* DOKS.**Fonte: Autoria própria**

LANÇAR E IMPLANTAR		MONITORAR E OTIMIZAR	
Lançamentos frequentes		Monitoramento de desempenho	
Lançamento automático		Monitoramento contínuo	
Planejamento de lançamento		Métricas de medição	
Implantação contínua		Feedback contínuo	
Implantação automatizada		Melhoria contínua	

(a) Avaliação das práticas e ferramentas do Ops no início do estudo.

LANÇAR E IMPLANTAR		MONITORAR E OTIMIZAR	
Lançamentos frequentes		Monitoramento de desempenho	
Lançamento automático		Monitoramento contínuo	
Planejamento de lançamento		Métricas de medição	
Implantação contínua		Feedback contínuo	
Implantação automatizada		Melhoria contínua	

(b) Avaliação das práticas e ferramentas do Ops no final do estudo.

Figura 26: Avaliação das práticas e ferramentas do Ops no *Framework* DOKS.**Fonte: Autoria própria**

4.1.4.3 GESTÃO DE CONHECIMENTO

Em relação à GC, a Figura 27 corresponde à (a) avaliação no início e à (b) avaliação no final do estudo. Pode-se observar que os processos de conversão de conhecimento “Externalização” e “Combinação” obtiveram avaliações melhores.

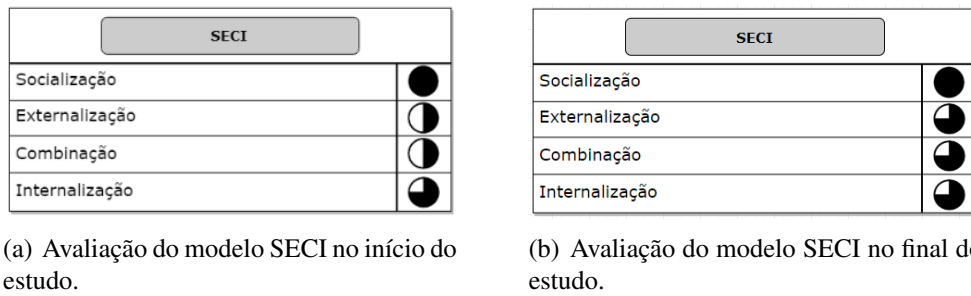


Figura 27: Avaliação da GC no *Framework* DOKS.

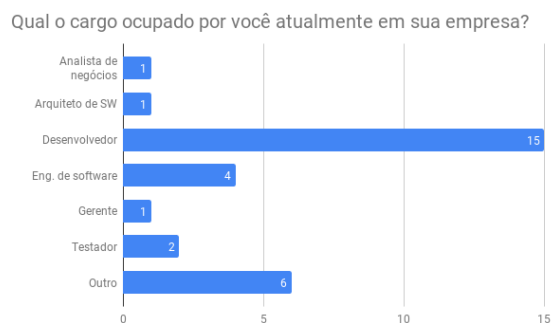
Fonte: Autoria própria

4.2 RESULTADOS DA SURVEY

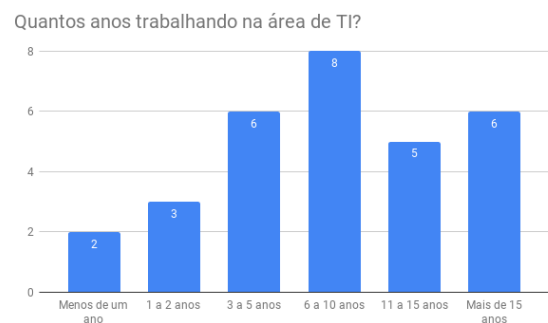
Esta seção possui os resultados obtidos na *Survey* realizada em âmbito nacional, conforme o planejamento na Seção 3.3. A *Survey* obteve um total de 30 respostas e para facilitar a visualização, os dados serão mostrados nas próximas seções conforme a estrutura definida para a *Survey* na Seção 3.3.2.

4.2.1 PERFIL DO PROFISSIONAL E DA EMPRESA

Na Figura 28 são exibidos os resultados referentes ao perfil do profissional, que corresponde à: (a) qual o cargo ocupado pelo participante na empresa que atualmente trabalha; e (b) quantos anos está no mercado de trabalho atuando na área de TI. Dos 30 participantes da *Survey*, metade são desenvolvedores e, no geral, são profissionais com um bom tempo de experiência na área de TI.



(a) Respostas sobre o cargo exercido pelo profissional.

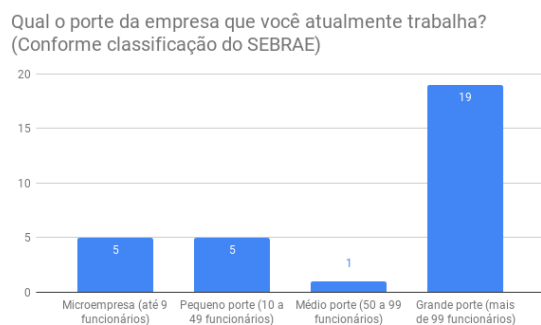


(b) Respostas sobre o tempo de trabalho na área de TI.

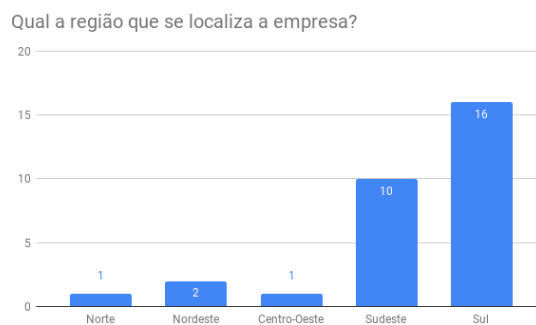
Figura 28: Gráficos sobre o perfil do profissional.

Fonte: Autoria própria

Na Figura 29 são exibidos os resultados referentes ao perfil da empresa, que corresponde à: (a) qual o porte da empresa na qual o respondente trabalha; e (b) qual a região que se localiza essa empresa. Diante das respostas, predomina-se profissionais que trabalham em empresas de Grande porte, conforme o SEBRAE ¹, e empresas localizadas na região Sul.



(a) Respostas sobre o porte da empresa.



(b) Respostas sobre a localização da empresa.

Figura 29: Gráficos sobre o perfil da empresa.

Fonte: Autoria própria

4.2.2 EXPERIÊNCIA COM DEVOPS

Na Figura 30 são exibidos os resultados referentes à experiência e conhecimento sobre a cultura DevOps do profissional: (a) apresenta o conhecimento do profissional sobre a cultura DevOps; e (b) experiência com a cultura DevOps. E a Figura 31 apresenta as respostas sobre a questão do participante saber se os colegas de trabalho têm conhecimento ou experiência com a cultura DevOps e suas práticas.

No cenário nacional, os profissionais têm adquirido mais conhecimento e experiência com aspectos e práticas da cultura, haja vista que dentre os 30 profissionais, 80% sabe o que é a cultura DevOps e 66,7% possui alguma experiência com o DevOps. Outro ponto importante a ser levantado, é que 43,3% opinam que seus colegas tenham conhecimento e experiência com o DevOps, o que beneficia e favorece a adoção efetiva da cultura.

¹ SEBRAE - <http://www.sebrae.com.br/>

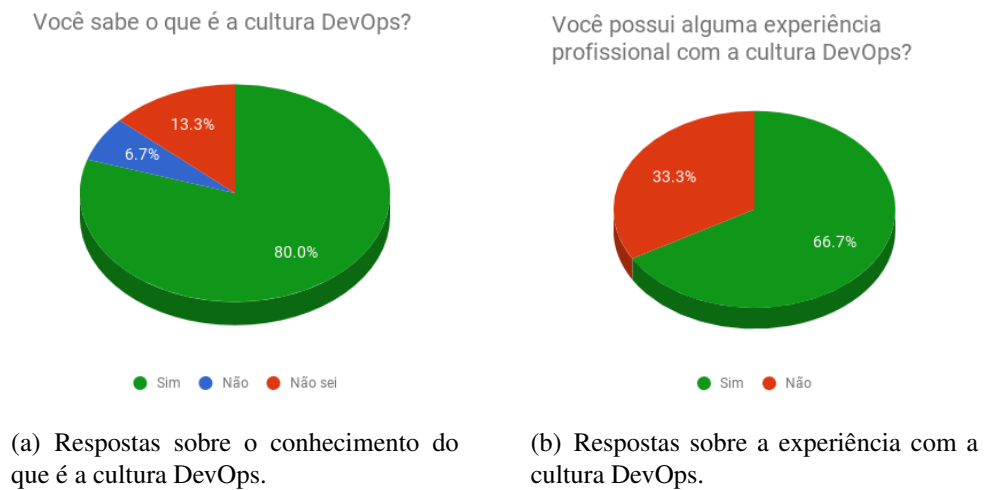


Figura 30: Gráficos sobre a experiência e conhecimento da cultura DevOps do profissional.

Fonte: Autoria própria

Em sua opinião, seus colegas de trabalho têm conhecimento ou experiência com a cultura DevOps e suas práticas?

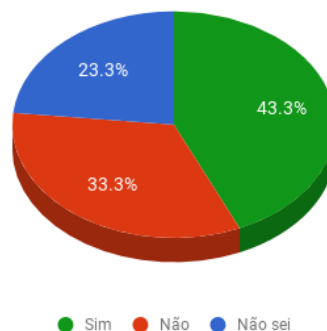


Figura 31: Respostas sobre a experiência dos colegas com o DevOps.

Fonte: Autoria própria

4.2.3 CULTURA DEVOPS NA EMPRESA

Na Figura 32 são exibidos os resultados referentes às perguntas que investigam (a) se a empresa atual do profissional utiliza práticas da cultura DevOps para o desenvolvimento de seus sistemas, (b) como é a caracterizada a cultura de trabalho e (c) se a empresa busca alternativas para automatizar etapas no processo de desenvolvimento. Para melhor visualizar as respostas da subfigura (b) da Figura 32, foi feita a legenda conforme o Quadro 12.

Quadro 12: Legenda da pergunta sobre a caracterização da cultura de trabalho da empresa.

Legenda	Resposta Original
Resposta 1	As equipes (desenvolvimento, operações, testes, controle de qualidade, etc) envolvidas no processo de desenvolvimento são independentes e com pouca colaboração
Resposta 2	As equipes (desenvolvimento, operações, testes, controle de qualidade, etc) envolvidas no processo de desenvolvimento são dependentes, mas com pouca colaboração e interação
Resposta 3	As equipes (desenvolvimento, operações, testes, controle de qualidade, etc) envolvidas no processo de desenvolvimento são colaborativas, compartilham responsabilidades e trocam conhecimento e experiências

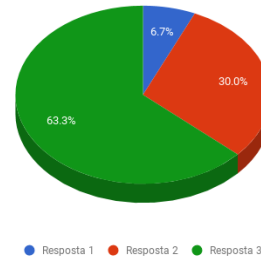
Fonte: Autoria própria

A empresa/equipe na qual você trabalha atualmente utiliza as práticas da cultura DevOps para o desenvolvimento de suas soluções e serviços?



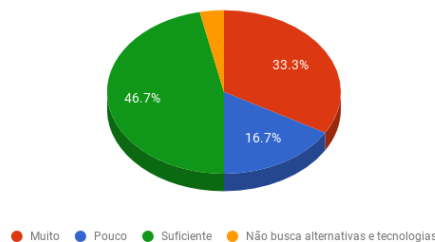
(a) Respostas sobre a empresa utilizar práticas da cultura DevOps para desenvolver seus sistemas.

O DevOps é um movimento cultural. A partir dessa afirmação, qual a resposta que melhor descreve a cultura de trabalho da sua empresa/equipe hoje?



(b) Respostas sobre à cultura de trabalho na empresa.

A empresa/equipe busca alternativas e tecnologias para automatizar etapas do processo de desenvolvimento?



(c) Respostas sobre a busca de alternativas para automatizar o processo de desenvolvimento.

Figura 32: Gráficos acerca da Cultura DevOps na empresa.**Fonte: Autoria própria**

A nível de empresa, pode-se considerar que a cultura DevOps está em um patamar bom de maturidade, pois dentre as respostas advindas dos profissionais, 70% afirmam que sua empresa aplica e trabalha com práticas da cultura para o desenvolvimento de suas soluções. Adicionalmente, a cultura de trabalho nessas empresas são propensas a facilitar a prática e

adoção do DevOps, dado que 63,3% das respostas apontam para equipes de desenvolvimento colaborativas e que compartilham responsabilidades e conhecimentos, além das empresas e equipes buscarem abordagens para automatizar etapas no processo de desenvolvimento.

Ainda relacionado a cultura DevOps na empresa, relacionado à prática de Implantação Contínua foi questionado a frequência de implantação realizada no sistema e como se comporta o sistema após a implantação na produção. A Figura 33 apresenta os seguintes resultados: (a) a frequência de implantação do sistema; e (b) o comportamento do sistema após a implantação.

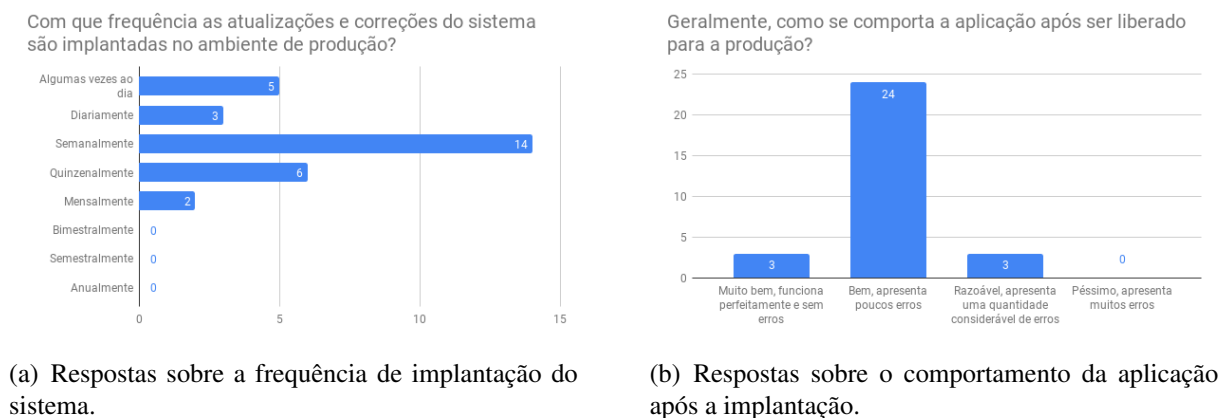


Figura 33: Gráficos acerca da Implantação Contínua.

Fonte: Autoria própria

Os resultados obtidos apontam que as empresas costumam implantar atualizações e correções de forma semanal, sendo algumas empresas que conseguem implantar mais de uma vez ao dia e os softwares são entregues com qualidade à produção, pois apresentam poucos erros. Esse aspecto está relacionado à importância da entrega do software de maneira rápida e frequente para o cliente, sendo considerado um diferencial competitivo.

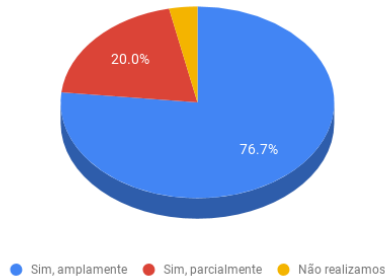
4.2.4 PRÁTICAS E FERRAMENTAS DEVOPS NA EMPRESA

Os resultados a seguir associam a realização de algumas das práticas do DevOps e as respectivas ferramentas que dão suporte para a execução de determinada prática.

A Figura 34 apresenta os seguintes resultados relacionados à controle de versão: (a) se o controle de versão é realizado na empresa; e (b) quais as ferramentas que a empresa utiliza para prover essa prática. O controle de versão é realizado por 76,7% das empresas e

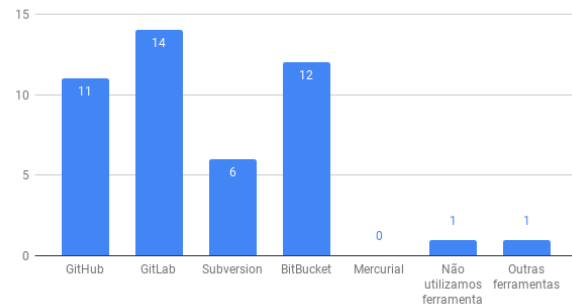
as ferramentas que são mais utilizadas para tal atividade são GitLab², GitHub³ e BitBucket⁴.

Sua equipe/empresa realiza Controle de Versão?



(a) Respostas sobre a realização do controle de versão.

Quais ferramentas são utilizadas para o Controle de Versão?



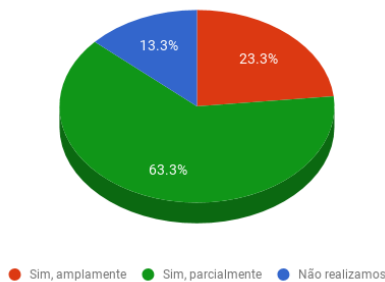
(b) Respostas sobre as ferramentas para o controle de versão.

Figura 34: Gráficos sobre o Controle de Versão da Empresa.

Fonte: Autoria própria

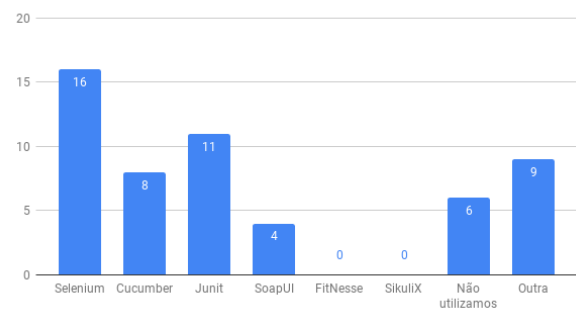
A Figura 35 apresenta os seguintes resultados relacionados a testes automatizados e contínuos: (a) se essa prática é realizada na empresa; e (b) quais as ferramentas que a empresa utiliza para prover essa prática. Os testes automatizados e contínuos ainda são praticados de forma parcial nas empresas (63,3%) e as ferramentas que se destacam para a realização dessa atividade são: Selenium⁵ e JUnit⁶.

Vocês realizam Testes Automatizados e Contínuos?



(a) Respostas sobre a realização de testes automatizados e contínuos.

Quais dessas ferramentas são utilizadas para os testes?



(b) Respostas sobre as ferramentas para os testes automatizados e contínuos.

Figura 35: Gráficos sobre os Testes Automatizados e Contínuos.

Fonte: Autoria própria

²GitLab - <https://gitlab.com/>

³GitHub - <https://github.com/>

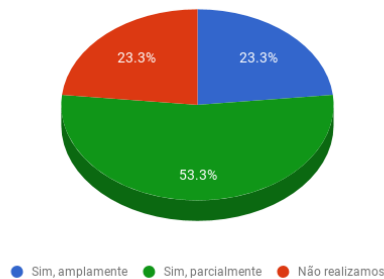
⁴BitBucket - <https://bitbucket.org/>

⁵Selenium - <https://www.seleniumhq.org/>

⁶JUnit - <https://junit.org/>

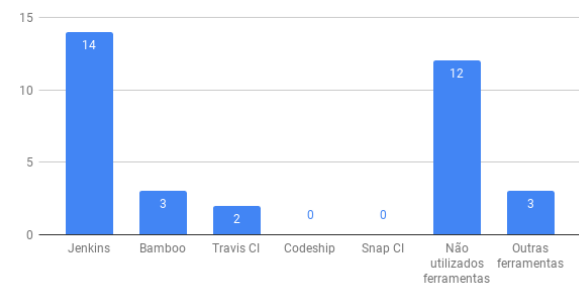
A Figura 36 apresenta os seguintes resultados relacionados à integração contínua: (a) se a integração contínua é realizada na empresa; e (b) quais as ferramentas que a empresa utiliza para prover essa prática. Conforme os resultados, a integração contínua ainda é uma atividade parcialmente realizada em 53,3% das empresas e a ferramenta mais utilizada para sua realização é o Jenkins⁷, contudo, 12 dos 30 participantes também não utilizam nenhuma ferramenta para sua execução.

Vocês realizam Integração Contínua?



(a) Respostas sobre a realização da integração contínua.

Quais dessas ferramentas são utilizadas para a Integração contínua?



(b) Respostas sobre as ferramentas para a integração contínua.

Figura 36: Gráficos sobre a Integração Contínua.

Fonte: Autoria própria

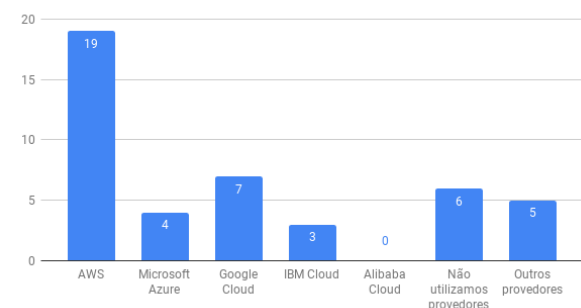
A Figura 37 apresenta os seguintes resultados relacionados aos serviços na nuvem: (a) se empresa utiliza serviços na nuvem; e (b) quais os provedores que a empresa utiliza para usufruir dos serviços na nuvem.

Vocês utilizam serviços na nuvem?



(a) Respostas sobre os serviços na nuvem.

Qual o provedor utilizado para prestar os serviços na nuvem?



(b) Respostas sobre os provedores de serviços na nuvem.

Figura 37: Gráficos sobre Serviços na Nuvem.

Fonte: Autoria própria

⁷Jenkins - <https://jenkins.io/>

O DevOps está ligado aos serviços na nuvem, no qual são disponibilizados servidores, armazenamento de dados, ambientes para programação e teste, entre outros serviços. As respostas indicam que 60,0% das empresas utilizam serviços na nuvem e o provedor desses serviços com mais destaque é o *Amazon Web Services* (AWS)⁸.

As respostas vistas na Figura 38 tem relação com a questão sobre outras práticas, ferramentas ou tecnologias que fornecem suporte à cultura DevOps e podem ser encontradas nos ambientes e equipes de trabalho de uma empresa. Para facilitar a visualização dos dados, o Quadro 13 apresenta a legenda das respostas pertencentes à pergunta.

Quadro 13: Legenda da pergunta sobre práticas, ferramentas ou tecnologias adicionais do DevOps.

Legenda	Resposta Original
Resposta 1	Implantação contínua
Resposta 2	Entrega contínua
Resposta 3	Implantação automatizada
Resposta 4	Monitoramento do desempenho do sistema
Resposta 5	Melhoria e reflexões contínua
Resposta 6	Comunicação entre as equipes
Resposta 7	Comunicação entre os indivíduos
Resposta 8	Compartilhamento do conhecimento entre indivíduos e equipes
Resposta 9	Ambientes similares ao da produção para teste e desenvolvimento
Resposta 10	Gestão dos requisitos
Resposta 11	União da equipe de desenvolvedores e a equipe de operações

Fonte: Autoria própria

Alguma dessas outras práticas, ferramentas ou tecnologias são encontradas nos ambientes e equipes de trabalho da sua empresa?

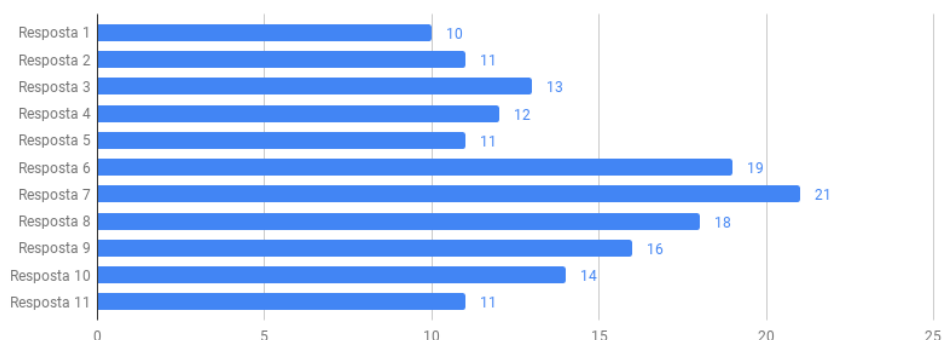


Figura 38: Gráfico sobre as práticas, ferramentas ou tecnologias adicionais que dão suporte para o DevOps.

Fonte: Autoria própria

⁸ Amazon Web Service - <https://aws.amazon.com/>

Todas as práticas, ferramentas ou tecnologias adicionais se mostram presentes de forma satisfatória nas equipes e empresas. O destaque fica para a comunicação e compartilhamento do conhecimento entre os indivíduos e equipes, aspecto importante para a adoção efetiva do DevOps e que é encontrado nos ambientes de trabalho das empresas.

Adicionalmente, foi incluída uma questão aberta para os participantes incluírem outras ferramentas ou serviços, além das que foram citadas nas perguntas anteriores, que auxiliam no processo de desenvolvimento e o Quadro 14 apresenta as respostas obtidas. O destaque fica para a ferramenta Jira que permite monitoramento de tarefas e acompanhamento de projetos e o uso de abordagens vistas em metodologias ágeis de desenvolvimento.

Quadro 14: Respostas sobre as práticas, ferramentas ou tecnologias adicionais que dão suporte para o DevOps.

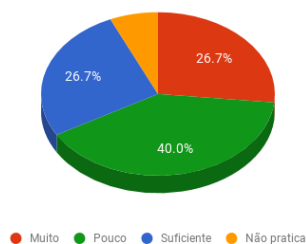
“Reuniões diárias e comunicação constante com clientes.”
“Utilizamos programação em pares, ciclos de testes fragmentados e revisão às cegas.”
“Utilizamos o scriptcase como ferramenta de desenvolvimento RAD com PHP e Javascript.”
“Não posso descrever. Ferramenta particular!”
“Jira.”
“Sonarqube, Docker e Ransher.”
“Jira - Utilizamos muito o Jira para controle de sprints; Confluence - para apoio entre os membros da equipe, como configurações de environments, documentações, manuais de instalação, reports; e Slack - comunicação da equipe; Tableau - <i>Business Intelligence</i> .”

Fonte: Autoria própria

4.2.5 GESTÃO DE CONHECIMENTO NA EMPRESA

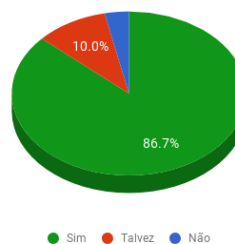
Na Figura 39 são apresentados os resultados em relação à GC na empresa: (a) se há o incentivo ou prática da GC na empresa; (b) se a GC é importante para o crescimento e evolução da empresa; e (c) em específico, se há a GC para aspectos relacionados ao DevOps. Já no Quadro 15 são apresentados as principais ações para GC elencadas pelos participantes que são realizadas na empresa.

Sua empresa incentiva ou pratica alguma forma de gestão do conhecimento?



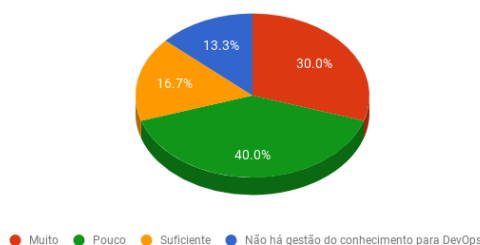
(a) Respostas sobre o incentivo ou prática da GC na empresa.

Na sua opinião, a gestão do conhecimento tem papel fundamental no crescimento e evolução da empresa?



(b) Respostas sobre a importância da GC para o crescimento da empresa.

Há a disseminação ou armazenamento do conhecimento referente à cultura DevOps e suas práticas?



(c) Respostas sobre a GC a respeito do DevOps.

Figura 39: Gráficos sobre aspectos gerais da GC na empresa.

Fonte: Autoria própria

A GC é vista como importante e fundamental para o crescimento e evolução da empresa por 86,7% dos profissionais. No entanto, o incentivo para a GC na empresa é predominantemente razoável e moderado, haja vista que não há muito incentivo por parte da empresa de acordo com 26,7% das respostas. Em particular, o conhecimento pertinente à cultura DevOps e suas práticas também não é disseminado e armazenado de forma ampla.

A GC é um aspecto relevante e importante para a adoção da cultura DevOps, apesar da GC ainda não ser praticada de forma abrangente dentro das organizações. Vale ressaltar que uma quantidade pequena, conforme os resultados, não pratica nenhuma ou incentiva nenhuma forma de GC. Isso indica que as empresas têm consciência do quão importante é compartilhar, armazenar e disseminar conhecimento entre indivíduos e equipes sobre processos, regras de negócio e ferramentas.

Dentre as principais ações realizadas pelas empresas voltadas para a GC, conforme apresentadas no Quadro 15, pode-se destacar: a disponibilização de cursos e treinamento para os funcionários, participação de workshops e eventos, execução de reuniões diárias e de planejamento, apresentações realizadas pelos próprios colaboradores sobre determinado tópico

para os demais e comunicação entre indivíduos, equipes e clientes. Dentre as respostas, também foram elencados algumas ferramentas que auxiliam na GC: repositórios de documentos, blogs, wikis e o uso de ferramentas como Trello⁹ e Ms SharePoint¹⁰.

Quadro 15: Respostas sobre as principais ações realizadas pela empresa voltadas para a GC.

“TalkCafé voluntário (apresentado por qualquer um da empresa quando quiser).”
“Cursos de repasse.”
“Blog e servidor de arquivos com manual e documento úteis.”
“A empresa possui um sistema para gestão de conhecimento, onde o mesmo é utilizado tanto por funcionários e clientes com informações sobre os sistemas desenvolvidos pela empresa. Também possui wikis e blogs internos.”
“Treinamentos”
“Além de vários workshops, estamos implantando um sistema para gestão e modelos KCS.”
“Nenhuma.”
“Compartilhamento de conhecimento entre os indivíduos.”
“Playbooks, documentações de processos, webinars, vídeos e tutoriais.”
“Muito pouco.”
“Workshops entre funcionários, cursos pela Udemey, entre outros, além de disponibilizar tempo do horário de trabalho para estudos.”
“Reuniões.”
“Até o momento nenhuma, mas está sendo iniciado um processo de documentação através de wiki.”
“Treinamentos.”
“Reuniões diárias de 5 minutos.”
“Reunião e disseminando a cultura dentro do processo como um todo pelos arquitetos.”
“Repositório de documentos.”
“Apenas comunicação oral e escrita, quando solicitado.”
“Workshops; Aprimoramento pessoal; intercâmbio profissional; e <i>job rotation</i> .”
“Integração de equipes multidisciplinares.”
“Documentação de APIs do sistema, de Incidentes, de visitas aos clientes, conhecimentos gerados pela equipe (individualmente e em grupo), repasses feitos internamente, workshops e eventos frequentados pelos membros da equipe.”
“Guildas com tópicos de interesse da empresa, exemplo: <i>data science</i> , <i>continuous integration</i> , DDD, etc. Minicursos ministrados pelos próprios funcionários para outros funcionários, sobre um tema que seja do interesse de duas pessoas ou mais.”
“Não sei dizer.”
“Ferramenta Wiki.”
“Ms SharePoint.”
“Trello.”
“Nenhuma.”
“Treinamentos internos, capacitação conjunta da equipe, um setor responsável somente por mapear pessoas e funcionários que possuem o domínio de certa tecnologia, Techleads de todas as tecnologias que possuem a responsabilidade de remover um eventual blocker de algum colaborador em uma sprint, por exemplo.”
“Documentação, testes, reuniões e aplicação de métodos.”

Fonte: Autoria própria

⁹Trello - <https://trello.com/>

¹⁰Ms SharePoint - <https://products.office.com/pt-br/sharepoint/collaboration>

4.2.6 BENEFÍCIOS NA ADOÇÃO DA CULTURA DEVOPS

Na Figura 40 são apresentados os benefícios com a adoção da cultura DevOps e no Quadro 17 são elencados benefícios e comentários adicionais. A fim de melhorar a visualização, o Quadro 16 apresenta a legenda das respostas da pergunta em questão.

Quadro 16: Legenda da pergunta sobre benefícios da cultura DevOps.

Legenda	Resposta Original
Resposta 1	Aquisição de clientes
Resposta 2	Retenção de clientes
Resposta 3	Mais qualidade no software desenvolvido
Resposta 4	Melhor participação no mercado
Resposta 5	Crescimento na receita/faturamento
Resposta 6	Maior produtividade no desenvolvimento
Resposta 7	Entregas melhores e frequentes do software
Resposta 8	Melhor clima organizacional
Resposta 9	Nenhum benefício

Fonte: Autoria própria

Quais os benefícios notados com a cultura DevOps?

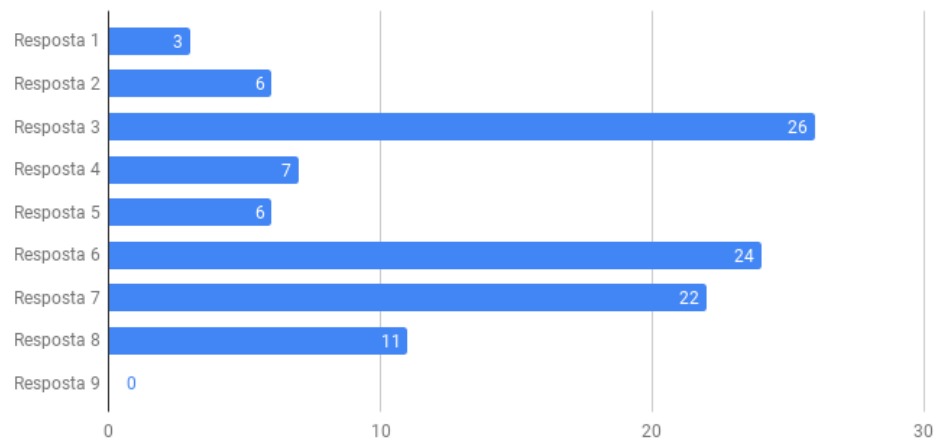


Figura 40: Gráfico sobre os benefícios da adoção da Cultura DevOps.

Fonte: Autoria própria

Todos os benefícios disponíveis como respostas foram marcados como existentes com a adoção da cultura DevOps, destacando-se “Mais qualidade no software desenvolvido”, “Maior produtividade no desenvolvimento” e “Entregas melhores e frequentes do software”. Importante destacar que a opção “Nenhum benefício” não foi escolhida nenhuma vez. Portanto, quando há a adoção da cultura DevOps existirá benefícios para a empresas e para as equipes, principalmente na qualidade e produção do software e a sua entrega ao cliente.

Quadro 17: Respostas sobre os benefícios e comentários adicionais da adoção da Cultura DevOps.

“Embora a empresa não tenha um projeto oficial para implantação de processos DevOps, temos caminhando nesse sentido com algumas ações que buscam fomentar essa cultura. A adoção do devops, além dos já citados nesta pesquisa, traz uma segurança maior para o QAS da empresa e facilita um modelo de intervenção antecipada (proativa) evitando que o cliente seja o primeiro a saber do bug.”
“Disseminação de conhecimento e transformação de profissionais em profissionais multidisciplinares.”
“Já trabalhei em uma empresa onde a cultura DevOps não era pregada e recentemente estou tendo a oportunidade em trabalhar em uma empresa onde a cultura de DevOps é amplamente pregada (inclusive com colaboradores que são contratados para desempenhar esse papel). Bom, o que eu posso dizer é que a diferença é gritante. A empresa no qual trabalho é uma multi-nacional e nota-se o porque dela ter crescido tanto e ter se tornado uma das maiores empresas brasileiras. O processo de desenvolvimento beira a excelência porque os funcionários são conscientizados da importância da cultura DevOps para a entrega de um produto. Enquanto que na antiga empresa o processo de desenvolvimento era lento e mal organizado, no meu atual emprego vejo um processo de desenvolvimento eficaz e ao mesmo tempo ágil.”

Fonte: Autoria própria

Nos comentários adicionais é notado a importância do DevOps para o desenvolvimento de software na atualidade e a contribuição que a cultura proporciona para a proatividade, a transformação de profissionais em multidisciplinares e o crescimento da organização no nível estrutural e de desenvolvimento de suas aplicações.

4.2.7 LIMITAÇÕES NA ADOÇÃO DA CULTURA DEVOPS

Na Figura 41 são apresentadas as limitações para a adoção da cultura DevOps e no Quadro 19 são elencadas limitações e comentários adicionais. A fim de melhorar a visualização, o Quadro 18 apresenta a legenda das respostas da pergunta em questão.

As limitações que mais impactam na adoção efetiva do DevOps são associadas à falta de conhecimento das práticas e das ferramentas disponíveis, à distância entre as equipes de Dev e as equipes de Ops, a baixa aderência dos colaboradores e pouco incentivo da alta gerência. Por conseguinte, constata-se a importância das ações da área de GC para a adoção da cultura e a aproximação das equipes de Dev e Ops, sendo este ponto o objetivo do DevOps, o que em muitas organizações não é possível ser realizado. Complementarmente, o engajamento dos colaboradores e da alta gerência é um aspecto fundamental para suprir barreiras e adotar de forma efetiva a cultura.

Os comentários adicionais destacam que os afazeres diários relacionados ao projeto, a

disseminação da cultura e a existência de equipes que são separadas, não são multifuncionais e não compartilham metas são fatores limitantes à adoção efetiva da cultura DevOps.

Quadro 18: Legenda da pergunta sobre limitações na adoção da cultura DevOps.

Legenda	Resposta Original
Resposta 1	Pouco incentivo da alta gerência
Resposta 2	Problemas nos ambientes (desenvolvimento/ teste)
Resposta 3	Questões financeiras
Resposta 4	Falta de conhecimento das práticas e ferramentas
Resposta 5	Conflitos entre os indivíduos da equipe/produto/empresa
Resposta 6	Pouca aderência dos colaboradores
Resposta 7	Limitação do processo de desenvolvimento
Resposta 8	Equipes de Dev e Ops distantes

Fonte: Autoria própria

Quais as limitações encontradas para a adoção do DevOps?

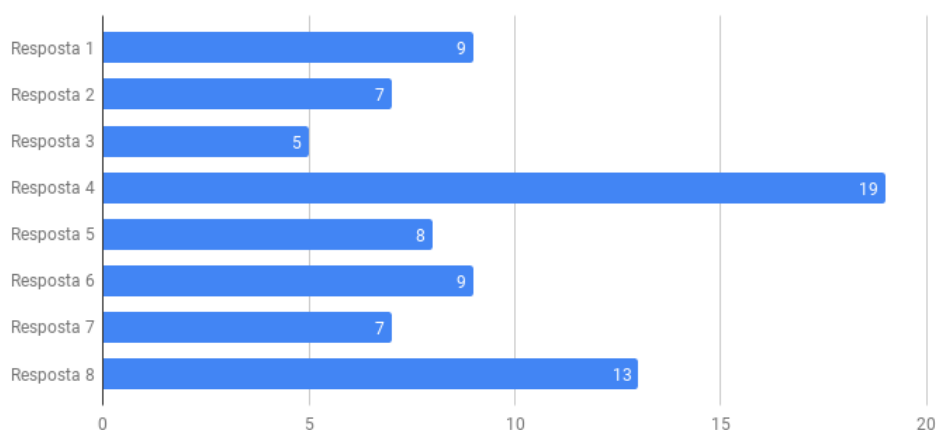


Figura 41: Gráfico sobre as limitações na adoção da Cultura DevOps.

Fonte: Autoria própria

Quadro 19: Respostas sobre as limitações e comentários adicionais na adoção da Cultura DevOps.

“Muitas atividades para fazer, devido aos prazos dos projetos.”
“Creio que ainda é uma questão cultural mesmo. As empresas brasileiras demoram, às vezes, a aderir movimentos desse tipo.”
“Basicamente, foi apenas o esforço necessário para aprender e disseminar a cultura DevOps, e conseguir implantar toda a pipeline de continuous integration/delivery.”
“Equipes fortemente agrupadas apenas pelas especialidades e com metas diretamente relacionadas às especialidades, sem necessidade de envolver outros times para atingir a meta.”

Fonte: Autoria própria

4.3 RESPOSTAS ÀS QUESTÕES DE PESQUISA E DISCUSSÕES

Nesta seção são apresentadas as respostas para as questões de pesquisa propostas para as avaliações empíricas dispostas no Capítulo 3 e realiza as discussões diante dos resultados apresentados no decorrer da presente seção. Para possibilitar uma melhor compreensão as discussões serão feitas na ordem de apresentação do resultado, iniciando pelo Estudo de Caso e posteriormente da *Survey*.

4.3.1 DISCUSSÕES SOBRE O ESTUDO DE CASO

O Estudo de Caso almejava alcançar os objetivos relativos ao estudo da GC para apoiar a adoção da cultura DevOps, aplicar o *framework DOKS* para a avaliação de elementos principais que compõe o DevOps e avaliar a efetividade do DOKS e uso da GC para apoiar a cultura DevOps, conforme o Quadro 2.

Isto posto, os recursos advindos da GC utilizados para apoiar a implementação do DevOps foram o blog e seminários (no contexto da Empresa A são chamados de Talk-café, pois ocorrem as apresentações juntamente com coquetel de café da manhã) e esses recursos, conforme os resultados, se mostraram efetivos para auxiliar na adoção da cultura DevOps.

Houve a aplicação do *Framework* Conceitual Doks para avaliação dos principais elementos que compõe a cultura DevOps no cenário de desenvolvimento da Empresa A. A efetividade do DOKS é comprovada pelos comentários realizados pelos funcionários da empresa durante a segunda reunião para avaliação do *framework*. Foi destacado que é importante evidenciar os problemas, as lacunas em algumas situações são difíceis de serem resolvidas, entretanto evidenciá-las já indica aonde deve ser melhorado e ajustado. Além disso, o *framework* foi visto como um norte e pode servir como um método de policiamento, onde poderá ajudar no planejamento e como melhorar em aspectos dentro da empresa.

4.3.1.1 QUESTÃO DE PESQUISA DO ESTUDO DE CASO

A questão de pesquisa do Estudo de Caso era pautada em validar a eficiência e eficácia do propósito que o DOKS detém, que é a avaliação e viabilização da adoção da cultura DevOps em organizações desenvolvedoras de software com foco na GC, conforme o disposto na Seção 3.2.

Por consequência das atividades realizadas e dos resultados obtidos, a questão de pesquisa do Estudo de Caso é respondida de forma completa e positiva, no qual o *Framework*

Conceitual DOKS pode ser aplicado de forma interna nas empresas, se configurando como um significativo e proveitoso instrumento de avaliação e viabilização da cultura DevOps com foco na GC.

4.3.2 DISCUSSÕES SOBRE A *SURVEY*

A *Survey* almejava alcançar os objetivos relativos a mapear as práticas DevOps em casos de sucesso de adoção em um nível nacional e à identificação de potenciais benefícios e limitações na adoção de DevOps em cenários práticos e visava responder as questões de pesquisa mencionadas na Seção 3.3.

Em razão da quantidade de respostas obtidas na *Survey* disponibilizada em nível nacional, algumas questões de pesquisa só foram respondidas de forma parcial. Para tanto, necessita-se da execução de mais estudos e ampliação da *Survey* para obter respostas integrais para tais questões de pesquisa.

4.3.2.1 QUESTÃO DE PESQUISA 1

De acordo com os resultados obtidos por meio da *survey*, pode-se definir que QP1 foi respondida de forma parcial, tendo em vista o número de respostas obtidas, entretanto apresenta uma visão inicial do DevOps em nível nacional. Assim sendo, a cultura DevOps vem sendo adotada pelas empresas nacionalmente e apresenta-se como fundamental para a obtenção de melhorias no desenvolvimento e crescimento da empresa.

4.3.2.2 QUESTÃO DE PESQUISA 2

Em relação à QP2, ela foi respondida de forma integral e percebe-se que todas as práticas são ampla ou parcialmente realizadas dentro das empresas, cada qual utilizando-se de ferramentas que colaborem para a execução dessas práticas. De modo complementar, o Quadro 20 relaciona algumas práticas do DevOps e as recomendações de ferramentas para a sua execução.

Quadro 20: Relação entre práticas DevOps e ferramentas para executá-las.

Práticas	Ferramentas
Controle de Versão	GitLab, GitHub e Bitbucket
Testes Automatizados e Contínuos	Selenium e JUnit
Integração Contínua	Jenkins
Serviços na Nuvem	Amazon Web Service (AWS)

Fonte: Autoria própria

4.3.2.3 QUESTÃO DE PESQUISA 3

A QP3 está relacionada com o cenário da GC em um nível nacional, tendo em vista o número de respostas obtidas na *Survey*, tal questão foi respondida de forma parcial, possibilitando uma visão inicial da GC em âmbito nacional. Portanto, pode-se definir que os fundamentos e práticas dessa área são adotados dentro das organizações, porém é realizada de forma acanhada e moderada, apesar de ser vista como importante e fundamental para a evolução da organização.

4.3.2.4 QUESTÃO DE PESQUISA 4

No tocante à QP4, que era voltada para as práticas de GC nas organizações, tal questão foi respondida de forma integral. As respostas obtidas pela *Survey* apresentam que as principais práticas voltadas para a GC realizadas nas empresas se baseiam em: realização de cursos e treinamentos; reuniões diárias e de planejamento; comunicação entre as equipes e clientes; e o uso de ferramentas como blogs e wikis.

4.3.2.5 QUESTÃO DE PESQUISA 5

Em relação aos benefícios do DevOps, em consonância com a QP5, pode-se declarar que a questão foi respondida de forma integral e indica notoriamente os benefícios da adoção da cultura. Desse modo, a cultura DevOps proveem benefícios principalmente voltados para capacitar uma melhor e maior produtividade no desenvolvimento de software com mais qualidade, além de proporcionar a entrega desses softwares com mais frequência.

4.3.2.6 QUESTÃO DE PESQUISA 6

A QP6, voltada para as limitações que impedem a adoção efetiva da cultura DevOps, também foi respondida de forma integral, apresentando as barreiras e obstáculos que devem

ser superados para a adoção efetiva da cultura DevOps. Assim sendo, as principais limitações encontradas para a adoção da cultura são relacionadas à ausência de conhecimento referentes às práticas e ferramentas, a distância entre as equipes e a falta de colaboração e incentivo dos funcionários e gerência.

4.4 AMEAÇAS À VALIDADE

Para potencializar e garantir a qualidade dos resultados do presente trabalho e, principalmente, dos resultados obtidos no estudo de caso, a presente seção levanta as possíveis ameaças à validade. Do mesmo modo são citadas as ações que auxiliam a mitigar tais ameaças. A seguir estão elencadas as ameaças, conforme classificação de Travassos et al. (2002), e as respectivas ações corretivas:

- Validade de construção: essa validade, segundo Travassos et al. (2002), considera os relacionamentos entre a teoria e a observação, ou seja, se o tratamento reflete a causa bem e o resultado reflete o efeito bem. À vista disso, considera-se que as ameaças ao presente estudo estariam ligadas há um má definição da teoria em relação à GC, fazendo com que os participantes se baseiem em suposições próprias e a má definição do processo para realização do estudo de caso. Diante disso, duas ações foram planejadas e executadas para evitar tais ameaças: (1) fornecer insumos que providenciam uma base teórica breve e essencial sobre a GC; (2) definir protocolo de estudo de caso para garantir a execução correta do processo;
- Validade interna: essa validade define se o relacionamento observado entre o tratamento e o resultado é causal, e não é resultado da influência de outro fator não controlado ou medido (TRAVASSOS et al., 2002). Por meio de uma análise, as ameaças internas estavam associadas à rotatividade dos funcionários da Empresa A no departamento no qual o estudo de caso foi realizado. Adicionalmente, mudanças no uso de ferramentas e práticas também poderiam sombrear resultados tanto de GC quanto de DevOps. Em relação à rotatividade dos funcionários, não há muito a ser feito, pelo motivo de não ser incumbência dos autores do presente trabalho a decisão da entrada e saída de colaboradores. Entretanto, para auxiliar a mitigar essa ameaça houve o monitoramento das mudanças nas equipes, sendo relatada tal mudança e descartada da amostra de dados coletada. O mesmo processo de monitoramento e relato foi realizado com as ferramentas ou práticas;
- Validade externa: define condições que limitam a habilidade de generalizar os resultados

de um experimento para a prática industrial (TRAVASSOS et al., 2002). Ameaças externas para o presente trabalho foi em virtude da realização do estudo em apenas uma empresa. A ação para evitar esse tipo de ameaça foi a construção detalhada e bem estruturada de um protocolo de estudo de caso, com intuito de garantir a replicabilidade em outros contextos nos mesmos modos e moldes do presente estudo; e

- Validade de conclusão: relacionada à habilidade de chegar a uma conclusão correta a respeito dos relacionamentos entre o tratamento e o resultado do experimento (TRAVASSOS et al., 2002). As ameaças à conclusão foram: (i) poucas equipes utilizadas durante o estudo de caso; (ii) pouco tempo de execução do *framework*; e (iii) poucas respostas obtidas na *Survey*. Para mitigar tais ameaças foi realizada a análise e interpretação dos resultados com determinado rigor experimental, como previsto na Seção 3. Em especial, para o estudo de caso, foi realizada a confrontação entre os resultados obtidos por meio de um processo de triangulação.

4.5 PUBLICAÇÕES

Nesta seção são apresentados as publicações geradas com base nos resultados obtidos pela pesquisa e faz um levantamento das possíveis futuras contribuições científicas advindas do presente estudo, com a finalidade de colaborar com a transferência e disseminação do conhecimento gerado tanto para outros acadêmicos quanto para a indústria.

A partir dos resultados obtidos com o Questionário 1 foi possível realizar a primeira contribuição do presente trabalho. Houve a publicação de um artigo (PAVAN et al., 2018) nos Anais da 2ª Escola Regional de Engenharia de Software (ERES), organizado pela Sociedade Brasileira de Computação (SBC) e realizado na UTFPR do câmpus de Dois Vizinhos. O trabalho abordava a detecção e proposição de melhorias em cenário DevOps de uma empresa real de desenvolvimento de software.

Dentre as contribuições futuras, espera-se, pelo menos, mais três submissões em eventos e congressos da ES: (i) a publicação dos resultados obtidos com o Questionário 2, que realiza a análise das equipes após a execução de melhorias propostas a partir do Questionário 1; (ii) a avaliação realizada com o *Framework* DOKS, mostrando a efetividade de tal instrumento para auxiliar avaliação e adoção da cultura DevOps; e (iii) os resultados da *Survey* que mostra um cenário inicial e parcial da cultura DevOps e da GC em um nível nacional.

5 TRABALHOS RELACIONADOS

Nesta seção são apresentados trabalhos relacionados com o presente estudo. Em particular, são citados trabalhos e estudos referentes a *frameworks* para processos de software e *frameworks* voltados para a área de GC. Não será citado nenhum trabalho referente ao DevOps, pois a literatura ainda é escassa de referências e trabalhos sobre *frameworks* voltados para a cultura DevOps.

5.1 FRAMEWORKS PARA PROCESSOS

O *framework* proposto por Mezzomo (2015) tem o intuito de fomentar a avaliação e melhoria do processo de desenvolvimento de software nas organizações. Para tanto, o autor realiza um mapeamento e integração das boas práticas dos modelos e normas de referências existentes para a melhoria dos processos, sendo os seguintes modelos e normas utilizados: (i) MR-MPS, (ii) CMMI; e (iii) norma ISO/IEC 12207. Uma visão do fluxo geral entre as macros atividades do *framework* é apresentada na Figura 42. Dentro das atividades macros de “Avaliar Processo” e “Melhorar Processo” existem atividades específicas para serem realizadas a fim de alcançar o propósito de avaliação e melhoria do processo.

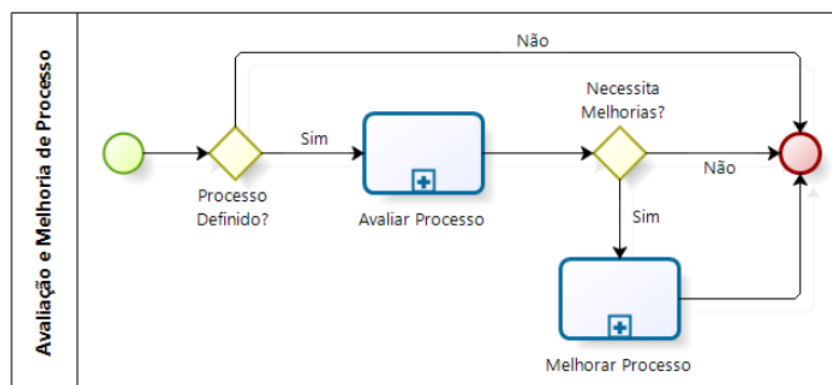


Figura 42: Visão macro do *framework* para avaliação e melhoria do processo de software.

Fonte: Anwar et al. (2017)

No trabalho de Souza (2013) também é proposto um *framework* de avaliação de processos de software. O *framework*, chamado de *framework* de avaliação integrada, define uma abordagem de avaliação de processo integrando práticas dos método de avaliação MA-MPS e SCAMPI A. Na Figura 43 é representado as fases do *framework* de avaliação integrada, no qual cada fase possui atividades próprias a serem realizadas para concluir a avaliação.

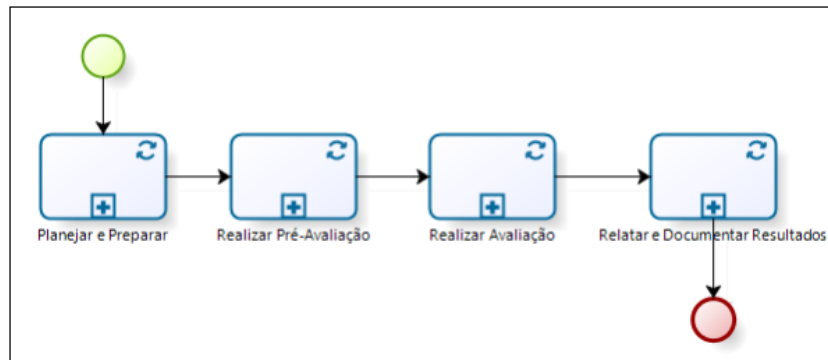


Figura 43: Representação das fases do *framework* de avaliação integrada.

Fonte: Anwar et al. (2017)

5.2 FRAMEWORKS PARA A GESTÃO DO CONHECIMENTO

O trabalho de Kavitha e Ahmed (2011) propõe um *framework* (Figura 44) para a captura e distribuição do conhecimento tácito gerado no uso de metodologias ágeis de desenvolvimento de software. O *framework* tem por objetivo o compartilhamento informal do conhecimento para membros da equipe que trabalham tanto localmente na empresa quanto distribuídos globalmente.

O *framework* parte do princípio que os colaboradores estão trabalhando em um ambiente colaborativo conectados pela internet/intranet. No que se refere à captura, armazenamento e disseminação do conhecimento entre os colaboradores, o *framework* descreve um conjunto de ferramentas integradas que dão suporte à essas atividades, tais como: fóruns de discussão, arquivos de e-mails, sessões informais de compartilhamento de conhecimento, respostas de questionários, notas de trabalho, aprendizado de projetos, entre outras.

As experiências geradas relacionadas à codificação, depuração, práticas, prova de conceito, entre outros, são inseridas por meio de uma interface pelo membro que são coletadas pelo Registrador de Experiência, as informações não estruturadas, como notas de trabalho e listas de tarefas relacionadas ao projeto, são armazenadas pelo Mapa de Ideias e as informações também podem ser postadas em um fórum de discussão. As informações coletadas pelo

registrador de experiência, mapa de ideias e fóruns podem ser estruturadas e armazenadas no repositório de conhecimento.



Figura 44: *Framework* proposto para a captura de conhecimento tácito em times de desenvolvimento ágeis.

Fonte: Adaptado de Kavitha e Ahmed (2011)

O trabalho de Anwar et al. (2017) tem por objetivo propor um *framework* conceitual para implementação do compartilhamento de conhecimento em organizações desenvolvedoras de software distribuídas globalmente. Para tanto, o *framework* (Figura 45) foi construído utilizando conceitos da psicologia e com o intuito de mitigar os fatores individuais, organizacionais, tecnológicos, culturais e geográficos que afetam o compartilhamento de conhecimento em equipes globais. O trabalho de Anwar et al. (2017) carece de validações para o *framework* proposto, sendo esse tópico um dos trabalhos futuros citados pelos autores.

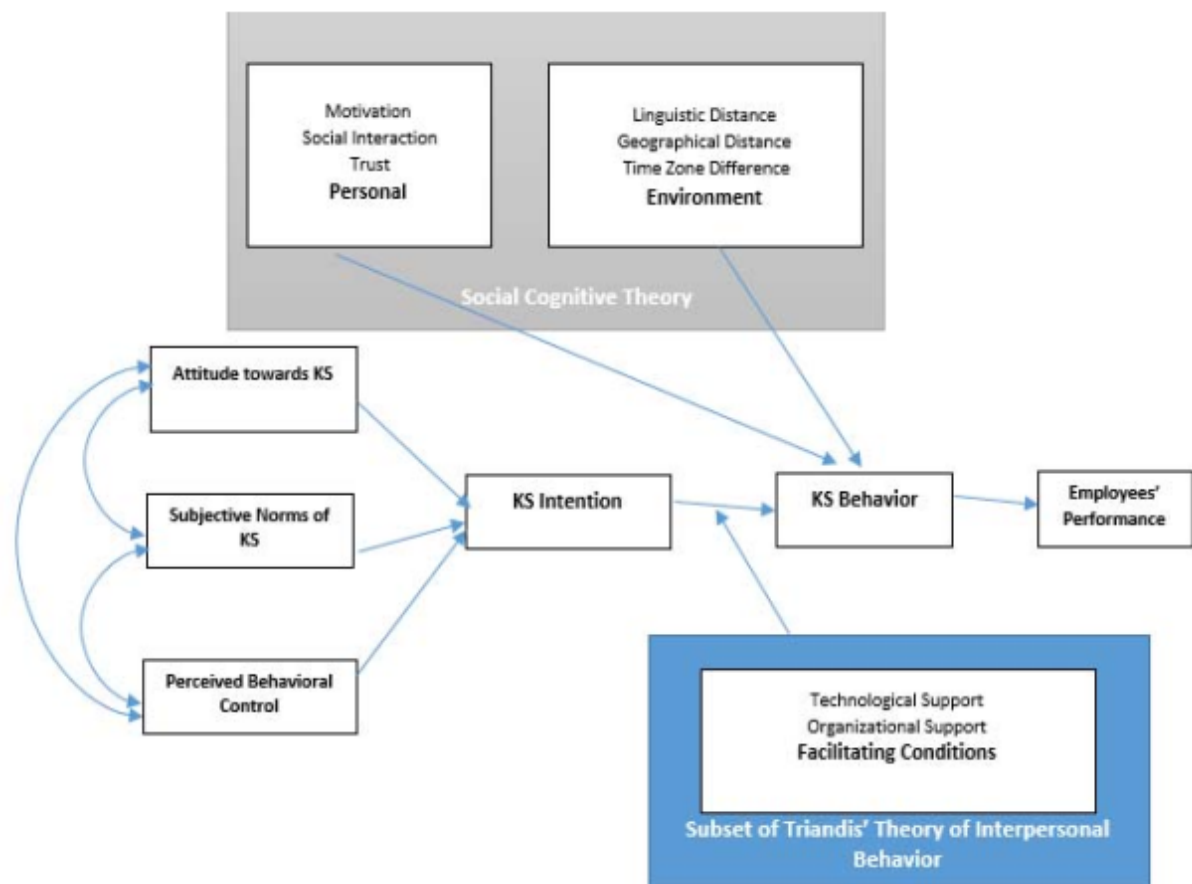


Figura 45: Representação do *Framework* para compartilhamento de conhecimento em organizações Globais de Desenvolvimento de Software.

Fonte: Anwar et al. (2017)

6 CONCLUSÃO

O DevOps é um movimento cultural que aproxima as equipes de desenvolvimento e as equipes de operações, tendo o apoio de ferramentas e práticas para prover a automação e a agilidade no desenvolvimento e entrega contínua de software com qualidade. Porém, a adoção efetiva da cultura DevOps é afetada pela ausência de estratégias advindas do meio acadêmico para o profissional, a ausência da GC nas organizações e a falta de consciência das organizações que o DevOps é um movimento cultural.

Desse modo, no decorrer do presente trabalho foram apresentados as diretrizes, procedimentos e os resultados obtidos com a execução da pesquisa sendo o objetivo principal: o fornecimento de um insumo de apoio à aplicação da prática DevOps em cenários práticos de desenvolvimento de software por meio do uso de recursos de GC. Complementarmente, realizar um mapeamento das práticas de DevOps no cenário nacional e identificar os benefícios e limitações na adoção da cultura.

Mediante os resultados apresentados, a aplicação do *Framework* Conceitual DOKS caracterizou-se como um insumo que apoia à adoção da Cultura DevOps com foco na GC em cenários práticos de desenvolvimento de software, por meio da avaliação de tópicos intrínsecos ao DevOps. Assim sendo, colaborou positivamente com a Empresa A que está em um estágio inicial de aplicação de práticas e princípios da cultura DevOps.

Adicionalmente, o presente trabalho apresentou uma visão inicial e parcial do cenário de DevOps e GC num âmbito nacional. Em particular, apontando as principais práticas do DevOps, bem como, os benefícios encontrados com a adoção da cultura e as limitações ao adotar a cultura DevOps. Em relação a GC, é visto que é uma área importante e precisa de mais incentivos e abordagens para ser executada de forma mais ampla e concreta dentro das empresas.

A partir das avaliações empíricas realizadas e resultados obtidos, as contribuições do presente estudo são: (i) o provimento de um instrumento para avaliação interna de aspectos e princípios do DevOps; (ii) levantamento inicial e parcial do cenário de DevOps

e GC a nível nacional; (iii) formação de recursos humanos, no qual os envolvidos no estudo empírico, bem como o condutor do projeto disseminaram e adquiriram conhecimento técnico e prático; (iv) publicação científica em um evento da área de ES, contribuindo para transferência e disseminação do conhecimento tanto para acadêmicos quanto para indústria; e (v) a aproximação entre as duas áreas de estudo do trabalho: DevOps e GC.

Por fim, como trabalhos futuros sugere-se: (i) a aplicação do estudo de caso em mais contextos reais de desenvolvimento; (ii) o aprimoramento e expansão da *Survey* para obtenção de mais respostas; (iii) a aplicação da *Survey* em um contexto internacional, como forma de comparação com o cenário nacional; e, (iv) como disposto na Seção 4.5, a realização de publicações científicas em eventos de ES. Adicionalmente, vislumbra-se a possibilidade de transferência tecnológica da indústria para a academia por meio de parcerias e integrações.

REFERÊNCIAS

- ALAVI, M.; LEIDNER, D. Knowledge management systems: Issues, challenges, and benefits. **Communications of the Association for Information Systems**, Atlanta, v. 1, n. 2, p. 37, 1999.
- ANWAR, R. et al. Conceptual framework for implementation of knowledge sharing in global software development organizations. In: **Proceedings of the 7th IEEE Symposium on Computer Applications Industrial Electronics (ISCAIE 2017)**. Langkawi: IEEE, 2017. p. 174–178.
- BASS, L. The software architect and devops. **IEEE Software**, v. 35, n. 1, p. 8–10, 2018.
- BECK, K. et al. **Manifesto for Agile Software Development**. 2001. Disponível em: <<http://agilemanifesto.org/>>. Acesso em: 01 de maio de 2018.
- COLOMO-PALACIOS, R. et al. A case analysis of enabling continuous software deployment through knowledge management. **International Journal of Information Management**, v. 40, p. 186 – 189, 2018.
- CRUZ, F. **Scrum e Agile em Projetos**: Guia completo. Rio de Janeiro: Brasport, 2015.
- DAVIS, J.; DANIELS, K. **Effective DevOps**: Building a culture of collaboration, affinity, and tooling at scale. 1. ed. Sebastopol: O'Reilly Media, Inc., 2016.
- DEBOIS, P. Agile infrastructure and operations: How infra-gile are you? In: **Proceedings of the Agile 2008 Conference (Agile 2008)**. Toronto: IEEE, 2008. p. 202–207.
- FARAHANI, F. F.; RAMSIN, R. Methodologies for agile product line engineering: A survey and evaluation. In: **Proceeding os the 13th International Conference on Intelligent Software Methodologies, Tools, and Techniques (SoMeT 2014)**. Langkawi: IOS International Publisher, 2014.
- FREITAS, H. et al. O método de pesquisa survey. **Revista de Administração da Universidade de São Paulo**, v. 35, n. 3, 2000.
- GOTTESHEIM, W. Challenges, benefits and best practices of performance focused devops. In: **Proceedings of the 4th International Workshop on Large-Scale Testing (LT 2015)**. New York: ACM, 2015. p. 3–3.
- HUMBLE, J.; FARLEY, D. **Continuous Delivery**: Reliable software releases through build, test, and deployment automation. Boston: Pearson Education, 2010.
- HUMBLE, J.; MOLESKY, J. Why enterprises must adopt devops to enable continuous delivery. **Cutter IT Journal**, v. 24, n. 8, p. 6–12, 2011.
- HUSSAIN, W.; CLEAR, T.; MACDONELL, S. Emerging trends for global devops: A new zealand perspective. In: **Proceedings of the 12th International Conference on Global Software Engineering (ICGSE 2017)**. Buenos Aires: IEEE, 2017. p. 21–30.

JABBARI, R. et al. What is devops?: A systematic mapping study on definitions and practices. In: **Proceedings of the Scientific Workshop Proceedings of XP2016 (XP 2016)**. New York: ACM, 2016. p. 1–11.

KAMUTO, M. B.; LANGERMAN, J. J. Factors inhibiting the adoption of devops in large organisations: South african context. In: **Proceedings of the 2nd International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT 2017)**. Bangalore: IEEE, 2017. p. 48–51.

KAVITHA, R. K.; AHMED, M. S. I. A knowledge management framework for agile software development teams. In: **Proceedings of the 2011 International Conference on Process Automation, Control and Computing (PACC)**. Coimbatore: IEEE, 2011. p. 1–5.

KIM, G. et al. **The DevOps Handbook**: How to create world-class agility, reliability, and security in technology organizations. Portland: IT Revolution Press, 2016.

MACHADO, P. H. de A. **Redução de desperdícios no desenvolvimento de software de grande porte por meio de ferramentas Lean**. 127 f. Dissertação (Mestrado em Engenharia de Produção e Sistemas) — Universidade Tecnológica Federal do Paraná, Pato Branco, 2017.

MEZZOMO, L. P. **Um framework para avaliação e melhoria do processo de software aderente ao CMMI, MR-MPS-SW e ISO/IEC 12207**. 125 f. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal do Pará, Belém, 2015.

NERUR, S. P.; MAHAPATRA, R.; MANGALARAJ, G. Challenges of migrating to agile methodologies. **Commun. ACM**, v. 48, n. 5, p. 72–78, 2005.

NIELSEN, P. A.; WINKLER, T. J.; NORBJERG, J. Closing the it development-operations gap: The devops knowledge sharing framework. In: **Proceedings of the 16th International Conference on Perspectives in Business Informatics Research (BIR 2017)**. Copenhagen: CEUR, 2017. p. 36–51.

NONAKA, I.; TAKEUCHI, H. **The Knowledge-creating Company**: How japanese companies create the dynamics of innovation. USA: Oxford University Press, 1995.

NONAKA, I.; TEECE, D. **Managing Industrial Knowledge**: Creation, transfer and utilization. London: SAGE Publications, 2001.

NONAKA, I.; TOYAMA, R.; KONNO, N. Seci, ba and leadership: a unified model of dynamic knowledge creation. **Long Range Planning**, v. 33, n. 1, p. 5–34, 2000.

O'LEARY, D. E. Enterprise knowledge management. **IEEE Computer**, v. 31, n. 3, p. 54–61, 1998.

PAVAN, D. R.; SOUZA Érica; OLIVEIRA, R. A. P. de. Detectando e propondo melhorias em cenários devops de empresas de desenvolvimento de software. In: **Anais da 2ª Escola Regional de Engenharia de Software (ERES 2018)**. Dois Vizinhos: SBC, 2018. p. 129–136.

PERERA, P.; SILVA, R.; PERERA, I. Improve software quality through practicing devops. In: **Proceedings of the 17th International Conference on Advances in ICT for Emerging Regions (ICTer 2017)**. Colombo: IEEE, 2017. p. 1–6.

PRESSMAN, R. **Engenharia de Software**: Uma abordagem profissional. 7. ed. New York: McGraw-Hill Education, 2011.

PUNTER, T. et al. Conducting on-line surveys in software engineering. In: **Proceedings of the 2nd International Symposium on Empirical Software Engineering (ISESE 2003)**. Rome: IEEE, 2003. p. 80–88.

PUPPET; DORA. **2015 State of DevOps Report**. 2015. Disponível em: <<https://puppet.com/resources/whitepaper/2015-state-devops-report>>. Acesso em: 27 de maio de 2018.

PUPPET; DORA. **2016 State of DevOps Report**. 2016. Disponível em: <<https://puppet.com/resources/whitepaper/2016-state-of-devops-report>>. Acesso em: 27 de maio de 2018.

PUPPET; DORA. **2017 State of DevOps Report**. 2017. Disponível em: <<https://puppet.com/resources/whitepaper/state-of-devops-report>>. Acesso em: 27 de maio de 2018.

RAZZAK, M. A.; AHMED, R. Knowledge sharing in distributed agile projects: Techniques, strategies and challenges. In: **Proceedings of the 4th Federated Conference on Computer Science and Information Systems (FedCSIS 2014)**. Warsaw: IEEE, 2014. p. 1431–1440.

RUNESON, P. et al. **Case Study Research in Software Engineering**: Guidelines and examples. 1. ed. New Jersey: Wiley Publishing, 2012.

SAMARAWICKRAMA, S. S.; PERERA, I. Continuous scrum: A framework to enhance scrum with devops. In: **Proceedings of the 17th International Conference on Advances in ICT for Emerging Regions (ICTer)**. Colombo: IEEE, 2017.

SATO, D. **DevOps na prática**: entrega de software confiável e automatizada. São Paulo: Casa do Código, 2017.

SAVOR, T. et al. Continuous deployment at facebook and oanda. In: **Proceedings of the 38th International Conference on Software Engineering Companion (ICSE-C 2016)**. Austin: IEEE, 2016. p. 21–30.

SCHLOSSNAGLE, T. Monitoring in a devops world. **Commun. ACM**, New York, v. 61, n. 3, p. 58–61, 2018.

SCHWABER, K.; SUTHERLAND, J. **The Scrum Guide**. 2017. Disponível em: <<https://www.scrumguides.org>>. Acesso em: 21 de novembro de 2018.

SEAMAN, C. B. Qualitative methods in empirical studies of software engineering. **IEEE Transactions on Software Engineering**, v. 25, n. 4, p. 557–572, 1999.

SERRADOR, P.; PINTO, J. Does agile work? — a quantitative analysis of agile project success. **International Journal of Project Management**, v. 33, n. 5, p. 1040–1051, 2015.

SHARMA, N.; SINGH, K.; GOYAL, D. P. Adoption of knowledge management practices in software engineering organizations: A survey of software engineers' perceptions. In: **Proceedings of the 2nd International Conference on Advanced Computing Communication Technologies (ACCT 2012)**. Rohtak: IEEE, 2012. p. 24–29.

SHARMA, S. **DevOps for Dummies**. USA: John Wiley & Sons, Inc, 2014.

SHARMA, S. **The DevOps Adoption Playbook: A guide to adopting devops in a multi-speed it enterprise**. 1. ed. Indianapolis: John Wiley and Sons, 2017.

SOMMERVILLE, I. **Software Engineering**. 9. ed. USA: Addison-Wesley Publishing Company, 2011.

SOUZA, E.; FALBO, R.; VIJAYKUMAR, N. Knowledge management initiatives in software testing: a mapping study. **Information and Software Technology**, v. 57, p. 378–391, 2014.

SOUZA, J. N. S. de. **Uma proposta de *framework* de avaliação de processos de software aderente ao MA-MPS e SCAMPI A**. 221 f. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal do Pará, Belém, 2013.

STAHL, D.; MARTENSSON, T.; BOSCH, J. Continuous practices and devops: beyond the buzz, what does it all mean? In: **Proceedings of the 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA 2017)**. Vienna: IEEE Conference Publishing Services, 2017. p. 440–448.

SWEBOK. **Guide of the Software Engineering Body of Knowledge**. 2014. Disponível em: <<http://swebok.org>>. Acesso em: 16 de maio de 2018.

TECHNOLOGIES, C. **Assembling the DevOps Jigsaw: Do you have all the right pieces in place?** 2015. Disponível em: <<https://www.ca.com/us/modern-software-factory/content/assembling-the-devops-jigsaw.html>>. Acesso em: 28 de maio de 2018.

TRAVASSOS, G. H.; GUROV, D.; AMARAL, E. A. G. do. **Introdução à Engenharia de Software Experimental**. Rio de Janeiro, 2002.

VADAPALLI, S. **Hands-on DevOps:: Explore the concept of continuous delivery and integrate it with data science concepts**. Birmingham: Packt Publishing, 2017.

VASANTHAPRIYAN, S.; TIAN, J.; XIANG, J. A survey on knowledge management in software engineering. In: **Proceedings of the International Conference on Software Quality, Reliability and Security (QRS 2015)**. Washington: IEEE Computer Society, 2015. p. 237–244.

VINCENZI, A. M. R.; DELAMARO, M.; MALDONADO, J. C. **Automatização de Teste de Software com Ferramentas de Software Livre**. Rio de Janeiro: Elsevier Editora Ltda., 2018.

WETTINGER, J.; ANDRIKOPOULOS, V.; LEYMAN, F. Automated capturing and systematic usage of devops knowledge for cloud applications. In: **Proceedings of the 3rd International Conference on Cloud Engineering (IC2E 2015)**. Tempe: IEEE Computer Society, 2015. p. 60–65.

WOHLIN, C. et al. **Experimentation in Software Engineering**. New York: Springer-Verlag Berlin Heidelberg, 2012.

XEBIALABS. **Periodic table of DevOps tools**. 2018. Disponível em: <<https://xebialabs.com/periodic-table-of-devops-tools/>>. Acesso em: 01 de junho de 2018.

APÊNDICE A – PROTOCOLO DE ESTUDO DE CASO

Neste Apêndice esta disposto o Protocolo de Estudo de Caso, sendo desenvolvido para auxiliar e servir de guia na execução das avaliações empíricas da presente pesquisa. Este documento ficará disponível para visualização, viabilizando a sua execução em outros contextos.

Protocolo de Estudo de Caso

1 Introdução e Contextualização

No cenário atual do mercado de software, as atualizações frequentes e contínuas dos sistemas tornaram-se prioridades em razão dos usuários e clientes esperarem novas funcionalidades e correções de defeitos o mais rápido possível [12]. Desse modo, agilizar e aumentar o número de entregas é um diferencial competitivo para as organizações de desenvolvimento de software e nesses casos a cultura DevOps (Desenvolvimento e Operações) favorece tal objetivo [7].

Desenvolvimento e Operações, em geral, apoiam as organizações de modo a criar uma cultura de colaboração, aproximando equipes e automatizando diversas fases dentro do processo para desenvolver um software. Entretanto, por se tratar de um tema relativamente novo, há uma falta de estratégias de adoção e *frameworks* para esse movimento cultural, tanto por pesquisadores quanto por profissionais da área [3] e, adicionalmente, o conhecimento e as experiências de DevOps estão praticamente disponíveis em larga escala, não sendo sistematicamente capturado e gerenciado [12].

Samarawickrama e Pereira [8] afirmam que existe um aumento da demanda por entregas rápidas de software, levando as empresas a crer que a solução é o DevOps, entretanto, as empresas desconhecem que o DevOps é um movimento cultural. Assim sendo, é comum que uma organização não adote o DevOps de maneira adequada, não estando alinhada às práticas e fundamentos advindos de tal cultura.

Em vista disso, um estudo de caso será realizado com o intuito de empregar um *framework* de apoio à aplicação da cultura DevOps em cenários práticos de desenvolvimento de software com foco na gestão de conhecimento. Essa aplicação visa garantir a adoção efetiva e ampla da cultura pela organização, possibilitando entregas e integrações contínuas do software. O estudo de caso irá compor um Trabalho de Conclusão de Curso e seguirá as diretrizes do presente Protocolo de Estudo de Caso.

Como forma de validar a eficácia do *framework*, vislumbra-se a realização de estudos empíricos no departamento de Novos Produtos de uma organização localizada no Sudoeste do Paraná. Este documento servirá de orientação para execução dos estudos empíricos e ficará disponível para visualização, viabilizando a sua replicação em outros contextos.

2 Background

A Gestão de Conhecimento (GC) tem como principal objetivo a criação, manutenção e transferência do conhecimento, com gerenciamento formal dos recursos de conhecimento para facilitar o acesso e reutilização do mesmo nas empresas [6, 11].

Na área de GC para o apoio de práticas de Engenharia de Software (ES) destacam-se os *frameworks*. Os *frameworks* de GC são uma estrutura ou conjunto de princípios, que são descritos de modo que forneçam a orientação em como conduzir a GC em uma organização, disponibilizando uma base teórica para executar alguma atividade ou ação [13].

Diversos *frameworks* e trabalhos sobre GC voltados para times, projetos e processos que utilizam os conceitos ágeis estão presentes na literatura [1, 2, 4]. Porém, para o movimento cultural DevOps a literatura ainda é escassa de referências que auxiliam na adoção efetiva da cultura com apoio da GC. Na revisão da literatura, o único trabalho encontrado que aborda

um *framework* conceitual específico para DevOps e utiliza a GC é apresentado no trabalho de Nielsen et al. [5]. O *framework* proposto em Nielsen et al., chamado de *DevOps Knowledge Sharing framework* (DOKS), tem como objetivo auxiliar na evolução do DevOps e avaliação de práticas e princípios da cultura em empresas e será detalhado a seguir.

O *framework* DOKS, representado na Figura 1, elenca os princípios e práticas do DevOps. O *framework* tem como base o modelo SECI (*Socialization, Externalization, Combination, Internalization*) para a GC, que tem como objetivo analisar e descrever as práticas de compartilhamento de conhecimento entre as equipes de desenvolvimento e operações em organizações que desenvolvem software. Sua função é auxiliar as empresas ou equipes que desejam evoluir em relação ao DevOps, avaliando o cumprimento dos principais elementos que compõe o DevOps.

Os três principais princípios identificados foram Agilidade, Colaboração e Integração. A respeito das práticas, o *framework* faz referência aos estágios do DevOps, que são: Medir e planejar (*Measure and plan*); Desenvolver e testar (*Develop and test*); Lançar e implantar (*release and deploy*); e Monitorar e otimizar (*Monitor and optimise*), mencionando as respectivas práticas de cada estágio. Além de utilizar o modelo SECI, para aumentar a conscientização sobre as diferentes maneiras de compartilhar o conhecimento, este sendo um importante aspecto do DevOps.

Para a validação do *framework* foi realizado uma pesquisa e entrevistas com os *stakeholders* de duas empresas, uma de pequeno porte e a outra de grande porte [5]. O feedback das empresas foi positivo, onde o *framework* ajudou a identificar os elementos para o DevOps, bem como elevou o foco das empresas para a importância desta área e, sobretudo, a ênfase no compartilhamento de conhecimento mostrou-se útil para explicitar os diferentes modos em que o conhecimento pode ser compartilhado entre as equipes e identificar medidas adequadas de melhoria [5].

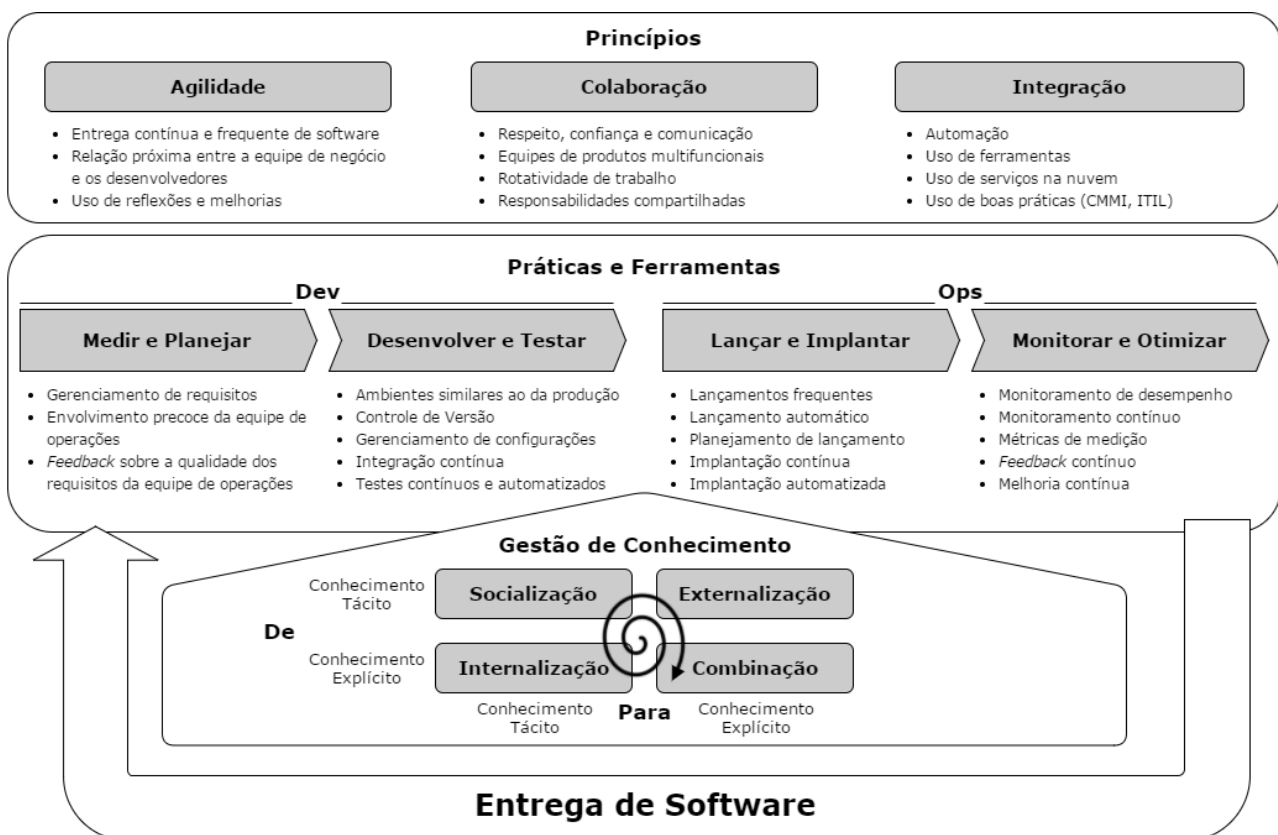


Figura 1: DOKS Framework proposto para auxiliar as empresas a evoluir em relação ao DevOps.

3 Projeto de estudo de caso

A seção a seguir destaca o objetivo do estudo, seus participantes, a forma de execução e detalha como serão realizadas as etapas de coleta, análise e validação dos dados e resultados. Adicionalmente, cita as possíveis limitações do estudo e como pretende-se expor os resultados obtidos.

3.1 Questão de Pesquisa

O *framework* conceitual DOKS proposto para a avaliação e viabilização da adoção da cultura DevOps em organizações desenvolvedoras de software com foco na GC é eficiente e eficaz ao seu propósito?

3.2 Participantes

As equipes que participarão do estudo de caso são do departamento de Novos Produtos de uma empresa de grande porte. A empresa está localizada no sudoeste do Paraná, é formada por mais de 500 colaboradores, fornece soluções para supermercados, lojas de materiais de construção e restaurantes, com nível de atuação nacional e conta com mais de 5 mil clientes. O departamento de Novos Produtos é responsável por desenvolver as novas soluções que a empresa oferecerá. As equipes que compõe tal departamento trabalham com metodologias ágeis e estão iniciando o trabalho com o DevOps. Ao todo participarão três equipes de um produto específico do departamento de Novos Produtos, o que totaliza 25 colaboradores.

3.3 Execução

O estudo de caso foi dividido em duas etapas: (i) Levantar dados iniciais das equipes acerca do DevOps e GC, por meio de questionário. Posteriormente, aplicar o *framework* DOKS com alguns integrantes das equipes e realizar a avaliação das diretrizes e princípios expostos no *framework*. Por fim, analisar os dados obtidos e propor melhorias; e (ii) Após a realização das melhorias, reavaliar as equipes acerca das melhorias propostas, bem como reaplicar o DOKS para validar se houve evolução em algumas diretrizes ou princípios. A Figura 2 ilustra o fluxo citado anteriormente.

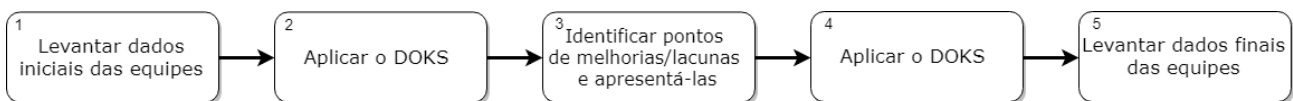


Figura 2: DOKS Framework proposto para auxiliar as empresas a evoluir em relação ao DevOps.

3.4 Coleta de dados

A coleta de dados ocorrerá prioritariamente no início e no final do estudo de caso, conforme a execução proposta. Para a coleta serão usadas algumas abordagens, dentre elas: questionários, o próprio *framework* DOKS e feedbacks obtidos da empresa. O intuito de utilizar tais abordagens é para atender ao objetivo do estudo de caso e avaliar a evolução das equipes em relação ao DevOps após a avaliação com o *framework* DOKS e a aplicação de melhorias para as lacunas identificadas. A seguir, são apresentadas de forma detalhada as abordagens usadas na pesquisa.

3.4.1 Questionários

Os questionários serão compostos de perguntas abertas e fechadas, e disponibilizados para os integrantes das equipes. Ao todo serão utilizados dois questionários: o primeiro questionário será utilizado para obter uma visão geral e inicial do DevOps e da GC nas equipes; e o segundo questionário será utilizado para levantar informações após a finalização do estudo e averiguar as mudanças que ocorreram. Os questionários foram disponibilizados para todos os integrantes das equipes. A seguir, serão apresentadas as estruturas e perguntas utilizadas nos questionários.

O Questionário 1, que faz parte da primeira parte do estudo de caso, tem o objetivo de fazer um levantamento inicial para obter uma visão geral da Cultura DevOps, juntamente com suas práticas e princípios, e como é realizada a GC nas equipes.

As perguntas que fazem parte do Questionário 1 seguem os seguintes tópicos: (1) levantamento do perfil do profissional; (2) histórico e experiência com o DevOps; (3) visão das práticas, ferramentas ou tecnologias executadas no ambiente e equipe de trabalho; e (4) como é percebido e praticado a GC nas equipes. As perguntas para cada tópico estão disponibilizadas a seguir:

- **Tópico 1 - Levantamento do perfil do profissional:**

- 1) Qual equipe você faz parte?
- 2) Qual cargo você ocupa atualmente?
- 3) Há quanto tempo você trabalha na área de TI?
- 4) Há quanto tempo você trabalha na empresa atual?

- **Tópico 2 - Histórico e experiência com o DevOps:**

- 1) Você já ouviu falar do termo DevOps?
- 2) Como você definiria o significado de DevOps? (caso não consiga definir deixe a resposta em branco)
- 3) Você já trabalhou com práticas da cultura DevOps?
- 4) Na sua opinião, a equipe/produto que atualmente você trabalha utiliza práticas da cultura DevOps?

- **Tópico 3 - Visão das práticas, ferramentas ou tecnologias executadas no ambiente e equipe de trabalho:**

- 1) Na sua opinião, quais dessas práticas, ferramentas ou tecnologias são encontradas no ambiente e equipe de trabalho?

- **Tópico 4 - como é percebido e praticado a GC nas equipes:**

- 1) Na sua opinião, há a GC sobre processos, negócios, tecnologias e ferramentas na(s) equipe(s) que você atualmente trabalha?
- 2) A empresa/equipe utiliza práticas e ferramentas que auxiliam na GC?
- 3) Na sua opinião, tais ferramentas são utilizadas adequadamente e frequentemente?
- 4) Gostaria de fazer alguma consideração sobre a aplicação de iniciativas de GC na empresa?

O Questionário 2, que faz parte da segunda parte do estudo de caso, tem o objetivo de fazer um levantamento final dos resultados. Esse questionário será executado após a aplicação das melhorias sugeridas para as lacunas encontradas na primeira parte do estudo. No presente estudo, as lacunas encontradas estavam relacionadas à: (i) Melhoria contínua e reflexão, (ii)

Aspecto Cultural do DevOps, (iii) Melhorias e incentivo da GC e (iv) Monitorar e Otimizar o sistema.

As perguntas que fazem parte do Questionário 2 seguem os seguintes tópicos: (1) identificação do profissional; (2) opinião sobre ações para melhorar o quesito de “Melhoria contínua e reflexão”; (3) opinião sobre ações para melhorar o quesito da “Cultura DevOps”; (4) opinião sobre ações para melhorar o quesito de GC; e (5) opinião sobre ações para melhorar o quesito de “Monitorar e Otimizar”. As perguntas para cada tópico estão disponibilizadas a seguir:

- **Tópico 1 - Identificação do profissional:**

- 1) Qual equipe você faz parte?
- 2) Qual cargo você ocupa atualmente?

- **Tópico 2 - Opinião sobre ações para melhorar o quesito de “Melhoria contínua e reflexão”:**

- 1) Em uma escala de 1 a 5, o quanto você concorda com a seguinte afirmação: as ações realizadas tiveram um efeito positivo para a melhoria contínua e reflexões nas equipes.
- 2) Gostaria de fazer alguma comentário, sugestão ou crítica sobre melhoria contínua e reflexões?

- **Tópico 3 - Opinião sobre ações para melhorar o “Aspecto Cultural do DevOps”:**

- 1) Em uma escala de 1 a 5, o quanto você concorda com a seguinte afirmação: a equipe/produto que atualmente você trabalha teve melhorias na cultura DevOps.
- 2) Gostaria de fazer alguma consideração, sugestão ou crítica em relação à cultura DevOps e as ações feitas?

- **Tópico 4 - Opinião sobre ações para melhorar o quesito de GC:**

- 1) Em uma escala de 1 a 5, o quanto você concorda com a seguinte afirmação: as ações realizadas tiveram um efeito positivo para melhorar a GC.
- 2) Em uma escala de 1 a 5, o quanto você concorda com a seguinte afirmação: o blog é uma ferramenta útil e boa para compartilhar o conhecimento.
- 3) Gostaria de fazer alguma consideração, sugestão ou crítica em relação ao blog e a GC?

- **Tópico 5 - Opinião sobre ações para melhorar o quesito de “Monitorar e Otimizar”:**

- 1) Em uma escala de 1 a 5, o quanto você concorda com a seguinte afirmação: a ação proposta pode ter um efeito positivo para melhorar o monitoramento e otimização do sistema.
- 2) Gostaria de fazer alguma comentário, sugestão ou crítica sobre a questão do monitoramento e otimização do sistema?

3.4.2 Aplicação do *Framework DOKS*

A estrutura do DOKS possibilita levantar dados de forma quantitativa para cada aspecto citado nos Princípios, Práticas e Ferramentas e GC. O DOKS será aplicado após a realização do primeiro questionário, no início do estudo, e ao final do estudo, antes do segundo questionário.

Para a avaliação dos tópicos apresentados no DOKS serão escolhidas algumas pessoas-chave dentre os 25 colaboradores que compõe as três equipes, devendo conter colaboradores que desempenham as funções referentes ao desenvolvimento, operação e negócio. A escolha de uma amostra dos integrantes das equipes foi realizada, pois se tornaria dispendioso para a empresa parar as três equipes do desenvolvimento do sistema e por serem peças importantes, eles possuem uma ampla visão das equipes e discernimento para realizar a avaliação de forma eficiente.

A etapa de avaliação utilizando o *framework* DOKS será mediante uma reunião com as pessoas-chave, onde será citado e explicado cada tópico do *framework* e em seguida, em consenso das pessoas-chave, será definido o nível que tal tópico é contemplado dentro das equipes que desenvolvem o produto. O nível que será escolhido para cada tópico seguirá o que é proposto no trabalho dos autores [5]:

- 1) Não cumprido (*Not fulfilled*)
- 2) Marginalmente cumprido (*Marginally fulfilled*)
- 3) Parcialmente cumprido (*Partially fulfilled*)
- 4) Largamente cumprido (*Largely fulfilled*)
- 5) Completamente cumprido (*Fully fulfilled*)

3.4.3 Feedback obtidos da Empresa

Além dos questionários e o *framework*, outra forma de coletar dados será recolhendo algumas informações e feedback da Empresa. Esta etapa se dará de maneira informal, por observação e durante as reuniões de aplicação do DOKS. Esse tipo de dado possibilita uma validação dos resultados obtidos com as outras abordagens.

3.5 Análise de dados

Simultaneamente com a coleta de dados, será realizada a análise dos dados de forma qualitativa e quantitativa. Os dados dos questionários serão sumarizados em gráficos e quadros para melhor visualização. Os dados obtidos com a execução do *framework* serão expostos em gráficos para facilitar a comparação dos resultados anteriores e posteriores à aplicação do DOKS.

Tais dados serão analisados por meio do método de Triangulação, que consiste em reunir diferentes tipos de evidências para apoiar uma proposição [9]. Portanto, os dados obtidos na primeira parte e na segunda parte do estudo, serão comparados para verificar se houve melhorias. Adicionalmente, informações e feedback obtidos da empresa serão contemplados na análise, para possibilitar uma validação dos resultados obtidos.

3.6 Validação do estudo

Com o propósito de validar o estudo de caso, será feito um levantamento das ameaças à validade ao estudo, seguindo a classificação de Travassos [10], e as ações que podem ser realizadas para evitá-las. Adicionalmente, serão cruzados os dados dos questionários, da execução do *framework* e o feedback sobre a realização do estudo na empresa por meio de um processo de triangulação.

3.7 Limitações do estudo

As possíveis limitações que podem ser encontradas durante o estudo e as respectivas ações para mitigá-las são:

- Quantidade de entrevistados: o questionário não terá um teor de obrigatoriedade, corre-se o risco de não obter participação integral dos participantes. Para mitigar essa limitação propõe-se a seguinte ação: enviar os questionários para as equipes e individualmente, por meio de ferramentas colaborativas/bate-papo que a empresa possui, explicando a importância da colaboração e solicitando a participação do colaborador;
- Colaboradores de férias e rotatividade nas equipes: alguns colaboradores poderão se ausentar durante a execução do estudo devido ao período de férias, bem como podem ocorrer demissões, chegada de novos colaboradores ou migração de colaboradores para outras equipes. Para mitigar essa limitação propõe-se as seguintes ações: (i) ao realizar qualquer atividade relacionada ao estudo introduzir e abordar o que está sendo feito, incluindo o colaborador dentro do contexto da pesquisa; e (ii) os colaboradores que migrarem para outras equipes fora do objeto de estudo serão descontados da amostra total de participantes, bem como aqueles que forem desligados da empresa; e
- Somente um contexto para aplicação do estudo: a pesquisa será realizada apenas em uma empresa real de desenvolvimento de software, limitando a pesquisa ao contexto de tal empresa. Para mitigar tal limitação propõe-se a seguinte ação: construção detalhada e bem estruturada do protocolo de estudo de caso, para possibilitar a execução em outros contextos.

3.8 Reportando resultados coletados

Espera-se que os resultados do estudo sejam divulgados por meio de publicações científicas da área de ES, contribuindo para a transferência tecnológica da academia para a indústria.

Referências

- [1] ANWAR, R., REHMAN, M., WANG, K. S., AMIN, A., AND AKBAR, R. Conceptual framework for implementation of knowledge sharing in global software development organizations. In *2017 IEEE Symposium on Computer Applications Industrial Electronics (ISCAIE)* (April 2017), pp. 174–178.
- [2] GERVIGNY, M. L. I., AND NAGOWAH, S. D. Knowledge sharing for agile distributed teams: A case study of mauritius. In *2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions) (ICTUS)* (Dec 2017), pp. 413–419.
- [3] KAMUTO, M. B., AND LANGERMAN, J. J. Factors inhibiting the adoption of devops in large organisations: South african context. In *Proceedings of the 2nd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)* (May 2017), pp. 48–51.
- [4] KAVITHA, R. K., AND AHMED, M. S. I. A knowledge management framework for agile software development teams. In *2011 International Conference on Process Automation, Control and Computing* (July 2011), pp. 1–5.
- [5] NIELSEN, P. A., WINKLER, T. J., AND NORBJERG, J. Closing the it development-operations gap: The devops knowledge sharing framework. In *BIR Workshops* (2017).

- [6] NONAKA, I., AND TEECE, D. *Managing Industrial Knowledge: Creation, Transfer and Utilization*. SAGE Publications, 2001.
- [7] PUPPET, AND DORA. 2017 state of devops report, 2017.
- [8] SAMARAWICKRAMA, S. S., AND PERERA, I. Continuous scrum: A framework to enhance scrum with devops. In *Proceedings of the Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer)* (Sept 2017).
- [9] SEAMAN, C. B. Qualitative methods in empirical studies of software engineering. *IEEE Transactions on Software Engineering* 25, 4 (Jul 1999), 557–572.
- [10] TRAVASSOS, G. H., GUROV, D., AND DO AMARAL, E. A. G. Introdução à engenharia de software experimental. Tech. rep., Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2002.
- [11] VASANTHAPRIYAN, S., TIAN, J., AND XIANG, J. A survey on knowledge management in software engineering. In *Proceedings of the IEEE International Conference on Software Quality, Reliability and Security (QRS)* (Aug 2015), pp. 237–244.
- [12] WETTINGER, J., ANDRIKOPOULOS, V., AND LEYMANN, F. Automated capturing and systematic usage of devops knowledge for cloud applications. In *Proceedings of the IEEE International Conference on Cloud Engineering (IC2E)* (2015), IEEE Computer Society, pp. 60–65.
- [13] WONG, K., AND ASPINWALL, E. Knowledge management implementation frameworks: A review. *Wiley InterScience* 11, 2 (2004), 93–104.

APÊNDICE B – *SURVEY* NA ÍNTEGRA

Neste Apêndice esta a estrutura em tópicos da *Survey* e suas respectivas perguntas.

Perguntas da *Survey*

A estrutura da *Survey* segue os seguintes tópicos: (1) levantamento do perfil do profissional e da empresa que trabalha; (2) conhecimento e experiência sobre o movimento cultural DevOps; (3) visão da cultura DevOps adotada e realizada na empresa; (4) práticas e ferramentas utilizadas na empresa que dão suporte e estão presentes na cultura DevOps; (5) visão sobre a Gestão de Conhecimento praticada na empresa; (6) benefícios notados com a adoção da cultura DevOps; e (7) limitações e desafios para a adoção da cultura. As perguntas para cada tópico estão disponibilizadas a seguir:

- **Tópico 1 - Levantamento do perfil do profissional e da empresa que trabalha:**

- 1) Qual o cargo ocupado por você atualmente em sua empresa?
- 2) Quantos anos trabalhando na área de TI?
- 3) Qual o porte da empresa que você atualmente trabalha? (Conforme classificação do SEBRAE)
- 4) Qual a região que se localiza a empresa?

- **Tópico 2 - Conhecimento e experiência sobre o movimento cultural DevOps:**

- 1) Você sabe o que é a cultura DevOps?
- 2) Você possui alguma experiência profissional com a cultura DevOps?
- 3) Em sua opinião, seus colegas de trabalho têm conhecimento ou experiência com a cultura DevOps e suas práticas?

- **Tópico 3 - Visão da cultura DevOps adotada e realizada na empresa:**

- 1) A empresa/equipe na qual você trabalha atualmente utiliza as práticas da cultura DevOps para o desenvolvimento de suas soluções e serviços?
- 2) O DevOps é um movimento cultural. A partir dessa afirmação, qual a resposta que melhor descreve a cultura de trabalho da sua empresa/equipe hoje?
- 3) A empresa/equipe busca alternativas e tecnologias para automatizar etapas do processo de desenvolvimento?
- 4) Com que frequência as atualizações e correções do sistema são implantadas no ambiente de produção?
- 5) Geralmente, como se comporta a aplicação após ser liberado para a produção?

- **Tópico 4 - Práticas e ferramentas utilizadas na empresa que dão suporte e estão presentes na cultura DevOps:**

- 1) Sua equipe/empresa realiza Controle de Versão?
- 2) Quais ferramentas são utilizadas para o Controle de Versão? (Marque todas que se aplicam)
- 3) Vocês realizam Testes Automatizados e Contínuos?

- 4) Quais dessas ferramentas são utilizadas para os testes? (Marque todas que se aplicam)
- 5) A integração contínua é a prática de mesclar todas as cópias de trabalho do desenvolvedor em uma linha principal compartilhada várias vezes ao dia. Vocês realizam Integração Contínua?
- 6) Quais dessas ferramentas são utilizadas para a Integração contínua? (Marque todas que se aplicam)
- 7) Vocês utilizam serviços na nuvem?
- 8) Qual o provedor utilizado para prestar os serviços na nuvem? (Marque todos que se aplicam)
- 9) Alguma dessas outras práticas, ferramentas ou tecnologias são encontradas nos ambientes e equipes de trabalho da sua empresa?
- 10) Vocês utilizam outras ferramentas ou serviços, além dos citados anteriormente, que auxiliam no processo de desenvolvimento? Se sim, cite-as e descreva a atividade que elas realizam.

- **Tópico 5 - Visão sobre a Gestão de Conhecimento praticada na empresa:**

- 1) Sua empresa incentiva ou pratica alguma forma de gestão do conhecimento?
- 2) Na sua opinião, a gestão do conhecimento tem papel fundamental no crescimento e evolução da empresa?
- 3) Há a disseminação ou armazenamento do conhecimento referente à cultura DevOps e suas práticas?
- 4) Quais são as principais ações que sua empresa realiza voltadas para a gestão do conhecimento?

- **Tópico 6 - Benefícios notados com a adoção da cultura DevOps:**

- 1) Quais os benefícios notados com a cultura DevOps?
- 2) Se houver necessidade, descreva outros possíveis benefícios encontrados com a Cultura DevOps.

- **Tópico 7 - Limitações e desafios para a adoção da cultura:**

- 1) Quais as limitações encontradas para a adoção do DevOps?
- 2) Se houver necessidade, descreva outras limitações que impediram, atrapalharam ou tiveram que ser superadas para adotar o DevOps.

ANEXO A – TABELA PERÍODICA DE FERRAMENTAS DEVOPS

[illegible]