

Stephany Mendes Oliveira

**Diretrizes para a Documentação de
Engenharia de Dados em uma abordagem
DevDocOps**

Sorocaba, SP

23 de Janeiro de 2024

Stephany Mendes Oliveira

Diretrizes para a Documentação de Engenharia de Dados em uma abordagem DevDocOps

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação (PPGCC-So) da Universidade Federal de São Carlos como parte dos requisitos exigidos para a obtenção do título de Mestre em Ciência da Computação. Linha de pesquisa: Engenharia de Software.

Universidade Federal de São Carlos – UFSCar

Centro de Ciências em Gestão e Tecnologia – CCGT

Programa de Pós-Graduação em Ciência da Computação – PPGCC-So

Orientador: Prof. Dr. Daniel Lucrédio

Sorocaba, SP

23 de Janeiro de 2024

Mendes Oliveira, Stephany

Diretrizes para a Documentação de Engenharia de Dados em uma abordagem
DevDocOps/ Stephany Mendes Oliveira. – 2024.2024

147 f. : 30 cm.

Dissertação (Mestrado) – Universidade Federal de São Carlos – UFSCar

Centro de Ciências em Gestão e Tecnologia – CCGT

Programa de Pós-Graduação em Ciência da Computação – PPGCC-So.

Orientador: Prof. Dr. Daniel Lucrédio

Banca examinadora: Prof. Dr. Daniel Lucrédio, Prof. Dr. Marcelo de Almeida Maia,

Prof. Dr. Auri Marcelo Rizzo Vincenzi

Bibliografia

1. Documentação Automática. 2. Engenharia de Dados. 3. DevDocOps. I.
Orientador. II. Universidade Federal de São Carlos. III. Título



UNIVERSIDADE FEDERAL DE SÃO CARLOS

Centro de Ciências em Gestão e Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Folha de Aprovação

Defesa de Dissertação de Mestrado da candidata Stephany Mendes Oliveira, realizada em 23/01/2024.

Comissão Julgadora:

Prof. Dr. Daniel Lucrédio (UFSCar)

Prof. Dr. Marcelo de Almeida Maia (UFU)

Prof. Dr. Auri Marcelo Rizzo Vincenzi (UFSCar)

O Relatório de Defesa assinado pelos membros da Comissão Julgadora encontra-se arquivado junto ao Programa de Pós-Graduação em Ciência da Computação.

*Dedico este trabalho ao meu grande amigo Me. Marco Antonio Gomes, por sempre
me incentivar a ir além,
ao Valter Luis Scarpel, meu pai, que sempre apoiou os meus voos e amparou os
meus saltos,
e por fim a mulher em mim que surgiu com esse processo.*

Agradecimentos

Agradeço,

aos meus pais que me apoiaram durante todas as adversidades e sempre incentivaram a dedicação aos estudos, a acreditar que todas as metas são possíveis de serem alcançadas.

ao meu grande amigo Me. Marco Antonio Gomes, por ter me indicado o programa de pós graduação da Ufscar Sorocaba, sem esta, o sonho de dedicar à pesquisa não existiria.

ao meu orientador, Dr. Daniel Lucrédio que me apoio, ensinou e direcionou durante todo o mestrado com sua gentileza, compreensão e inteligência.

ao Francisco Cosmo de Lima pelo valioso apoio na condução deste projeto, e por acreditar na minha capacidade de tirá-lo do papel.

ao Pedro Emilio Santos por sua confiança no potencial da ideia.

aos voluntários envolvidos na pesquisa, por se disponibilizarem a ajudar e contribuir com os resultados.

à Angélica Cunha dos Santos, pela parceria e amizade, por estar comigo durante todo o processo.

à banca avaliadora, composta pelos professores Dr. Marcelo de Almeida Maia Conte e Dr. Auri Marcelo Rizzo Vincenzi, pela gentileza, e por terem aceitado contribuir com esta pesquisa.

a todos os professores do PPGCC-So, que contribuíram com o meu crescimento e aperfeiçoamento acadêmico.

a todos que, direta ou indiretamente contribuíram com este objetivo.

Este trabalho foi realizado com o suporte da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

*“Ao comparar o dia de hoje com o de ontem,
aquele que avançou um único passo ou milímetro,
já é vitorioso.”
(Seja a Esperança, Daisaku Ikeda)*

Resumo

A dissertação apresenta uma investigação sobre a integração de práticas ágeis de desenvolvimento de software, como Integração Contínua (CI) e Desenvolvimento Contínuo (CD), com a filosofia *DevOps*, que busca a colaboração entre equipes de desenvolvimento e operações. A pesquisa explora o contexto da Engenharia de Dados, que lida com o desenvolvimento, implementação e manutenção de sistemas e processos para transformação de dados brutos em informações de alta qualidade.

O estudo destaca a importância da documentação técnica de software e dados em um ambiente de desenvolvimento ágil e como a automação desse processo pode melhorar a qualidade e a eficiência das entregas de software. A abordagem *DevDocOps* é introduzida como uma forma de integrar a documentação contínua no ciclo de vida de desenvolvimento de software, abordando os desafios de equilibrar documentação adequada com comunicação eficaz.

Além disso, a pesquisa investiga a automação da documentação de modelos de dados desenvolvidos no SAP HANA, um Sistema de Gerenciamento de Banco de Dados (SGBD) em memória. A documentação de *Calculation Views*, um objeto-chave no SAP HANA, é explorada em detalhes, destacando a importância da documentação precisa para o entendimento e uso eficaz desses modelos.

O estudo também compara várias ferramentas de automação de documentação de dados disponíveis no mercado e discute suas limitações em relação à abordagem *DevDocOps*. Conclui que, embora essas ferramentas ofereçam soluções eficientes para a gestão de dados e documentação, muitas vezes não atendem completamente às necessidades de integração contínua da documentação no ciclo de vida de desenvolvimento.

Para comprovar a aplicação prática da abordagem proposta, foi conduzido um estudo experimental onde as descobertas da literatura foram aplicadas em um ambiente de desenvolvimento real. Neste estudo, a abordagem *DevDocOps* foi implementada para integrar a documentação contínua no ciclo de vida de desenvolvimento de

engenharia de dados, com foco na automação do processo de documentação de modelos de dados do SAP HANA.

Por fim, a dissertação enfatiza a importância de uma abordagem holística para a documentação técnica em ambientes de desenvolvimento ágil, especialmente na Engenharia de Dados, e propõe diretrizes para a implementação bem-sucedida da abordagem *DevDocOps*. O estudo contribui para o avanço do conhecimento sobre a integração de práticas ágeis, *DevOps* e documentação técnica em ambientes complexos de desenvolvimento de software e engenharia de dados.

Palavras-chaves: Documentação Automática. Entrega Contínua (CD). Engenharia de Dados. Documentação Técnica. DevDocOps.

Abstract

The dissertation presents an in-depth investigation into the integration of agile software development practices, such as Continuous Integration (CI) and Continuous Deployment (CD), with the DevOps philosophy, which seeks collaboration between development and operations teams. The research explores the context of Data Engineering, which deals with the development, implementation, and maintenance of systems and processes for transforming raw data into high-quality information.

The study highlights the importance of technical documentation of software and data in an agile development environment and how automating this process can improve the quality and efficiency of software deliveries. The DevDocOps approach is introduced as a way to integrate continuous documentation into the software development lifecycle, addressing the challenges of balancing adequate documentation with effective communication.

Furthermore, the research investigates the automation of documentation for data models developed in SAP HANA, a powerful in-memory Database Management System (DBMS). Documentation of Calculation Views, a key object in SAP HANA, is explored in detail, emphasizing the importance of accurate documentation for understanding and effective use of these models.

The study also compares various data documentation automation tools available in the market and discusses their limitations in relation to the DevDocOps approach. It concludes that while these tools offer efficient solutions for data management and documentation, they often do not fully meet the needs of continuous documentation integration into the development lifecycle.

To prove the practical application of the proposed approach, an experimental study was conducted where the findings from the literature were applied in a real development environment. In this study, the *DevDocOps* approach was implemented to integrate continuous documentation into the data engineering development

lifecycle, with a focus on automating the SAP HANA data model documentation process.

Finally, the dissertation emphasizes the importance of a holistic approach to technical documentation in agile development environments, especially in Data Engineering, and proposes guidelines for the successful implementation of the DevDocOps approach. The study contributes to advancing knowledge on the integration of agile practices, DevOps, and technical documentation in complex software development and data engineering environments.

Key-words: Documentation Automation. Continuous Deployment (CD). Data Engineering. Technical Documentation. DevDocOps.

Lista de ilustrações

Figura 1 – Exemplo simplificado de estrutura de Calculation View	41
Figura 2 – Documento automático gerado no caso de detalhes do modelo .	46
Figura 3 – Relacionamento entre as Metas e Perguntas de Pesquisa com os Desenvolvedores.	57
Figura 4 – Relacionamento entre as Metas e Perguntas de Pesquisa com os Líderes	58
Figura 5 – Demonstração de Entrada no arquivo de configuração 'view_columns'	64
Figura 6 – Demonstração de Saída do arquivo <i>Markdown</i> de Dicionário de Dados	65
Figura 7 – Demonstração de Entrada do arquivo XML de <i>Calculation Views</i>	66
Figura 8 – Demonstração de Saída do arquivo <i>Markdown</i> de <i>Calculation Views</i>	67
Figura 9 – Exibição de Documentação Disponibilizada na Wiki	71
Figura 10 – Relacionamento entre as Metas, Perguntas de Pesquisa e Métricas na perspectiva dos Desenvolvedores.	75
Figura 11 – Relacionamento entre as Metas, Perguntas de Pesquisa e Métricas na perspectiva do Líder	75
Figura 12 – Avaliação da Facilidade de Utilização da Plataforma	86

Lista de tabelas

Tabela 1 – Comparação de Artefatos entre Equipes	50
Tabela 2 – Mapeamento de Artefatos de Documentação de Dados	51
Tabela 3 – Estabelecimento de Metas	55
Tabela 4 – Perspectiva do Desenvolvedor - métodos de extração de métricas	59
Tabela 5 – Perspectiva do Líder e métodos de extração de métricas	60
Tabela 6 – Perspectiva de Análise Comparativa e métodos de extração de métricas	60
Tabela 7 – Plano de Testes	61
Tabela 8 – Análise do Tempo Gasto na Documentação por Tipo e Desen- volvedor (GQM: Q1)	77
Tabela 9 – Estimativa de Tempo Gasto na Documentação por Tipo e De- senvolvedor (GQM: Q3)	78
Tabela 10 – Resultados da Avaliação da Presença de Documentação nos Objetos em Desenvolvimento (GQM Q4)	80
Tabela 11 – Resultados da Necessidade de Consulta à Documentação Ante- rior (GQM Q5)	81
Tabela 12 – Sincronização entre Documentação Anterior e Versão em Produ- ção (GQM Q6)	83
Tabela 13 – Facilidade de Acesso à Documentação (GQM Q7)	84

Lista de abreviaturas e siglas

CI	Integração Contínua
CD	Desenvolvimento Contínuo
AIOps	Inteligência Artificial e Operações
Dev	Desenvolvimento
Ops	Operações
MLOps	Machine learning e Operações
DataOps	Ciência de Dados e Operações
COMPDEC	Conferência Internacional de Engenharia de Dados Computacionais
UI	Interface do Usuário
ETL	Extração, Transformação e Carga
XML	Extensible Markup Language
CSV	Comma-separated values

Sumário

1	INTRODUÇÃO	27
1.1	Contextualização e Motivação	27
1.2	Objetivos	29
1.3	Metodologia	30
1.4	Contribuições	31
1.5	Organização do Trabalho	32
2	FUNDAMENTOS E TRABALHOS RELACIONADOS	33
2.1	Fundamentos	33
2.1.1	Integração Contínua (CI) e Desenvolvimento Contínuo(CD)	33
2.1.2	DevOps	34
2.1.3	Engenharia de Dados	36
2.1.4	Documentação Técnica de software e Dados	37
2.1.5	<i>Dev Documentation Ops</i>	38
2.1.6	Modelos de dados desenvolvidos no SAP HANA	39
2.2	Trabalhos Relacionados	42
3	ESTUDO EXPERIMENTAL	47
3.1	Introdução	47
3.2	Planejamento	47
3.2.1	Etapa A - Revisão Bibliográfica	48
3.2.2	Etapa B - Análise e Entendimento do ambiente organizacional no domínio da pesquisa	49
3.2.3	Etapa C - Definição dos parâmetros de teste	53
3.3	Execução	62
3.3.1	Etapa A - Desenvolvimento de <i>scripts</i> de automatização de documentação	63
3.3.2	Etapa B - Definição e construção do fluxo <i>DevDocOps</i>	67
3.3.3	Etapa C - Testes e Condução do Experimento	72

3.4	Resultados	74
3.4.1	Análise das Respostas dos Desenvolvedores ao <i>DevDocOps</i> : Uma Avaliação Detalhada	76
3.4.2	Análise das Respostas do Líder ao <i>DevDocOps</i> : Uma Avaliação Detalhada	90
3.4.3	Métricas e Análise de Resultados	95
3.4.4	Análise do Feedback dos Voluntários sobre a Utilização da Plataforma	100
3.4.5	Conclusão do Experimento	104
4	DIRETRIZES PARA A DOCUMENTAÇÃO DE ENGENHARIA DE DADOS EM UMA ABORDAGEM DEVDOCOPS .	105
	Diretrizes para a Documentação de Engenharia de Dados em uma abordagem DevDocOps	105
5	CONCLUSÃO	111
5.1	Principais Descobertas	111
5.2	Implicações Práticas	112
5.2.1	Versatilidade e Adaptabilidade do Modelo	112
5.2.2	Otimização de Tempo e Eficiência de Tarefas	112
5.2.3	Conclusão das Implicações Práticas	113
5.3	Recomendações para Futuras Pesquisas	113
5.3.1	Avaliação Investigativa de Melhoria dos <i>Templates</i> Padronizados de Documentação	113
5.3.2	Conclusão Final das Recomendações	114
5.4	Conclusão Final	114
	Referências	115
	APÊNDICE A – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO	119
	APÊNDICE B – QUESTIONÁRIO DE UTILIZAÇÃO DO DEVDOCOPS - DESENVOLVEDORES . .	123

APÊNDICE C – QUESTIONÁRIO DE UTILIZAÇÃO DO DEVDOCOPS - LÍDER	131
ANEXO A – SCRIPT DE GERAÇÃO DE DOCUMENTA- ÇÃO DE DICIONÁRIO DE DADOS	137
ANEXO B – SCRIPT DE GERAÇÃO DE DOCUMENTA- ÇÃO DE <i>CALCULATION VIEW</i>	139
ANEXO C – ARQUIVO DE CONFIGURAÇÃO PARA EXE- CUÇÃO - <i>CONFIG_DOC</i>	143
ANEXO D – PIPELINE DE DESENVOLVIMENTO E INTE- GRAÇÃO CONTÍNUA DE DOCUMENTAÇÃO	145

1 Introdução

Neste capítulo será apresentado o contexto no qual este trabalho está inserido e a motivação que originou esta pesquisa. Serão apresentados também os objetivos que foram alcançados, as principais contribuições, e por fim a organização desta monografia.

1.1 Contextualização e Motivação

O processo de desenvolvimento de software é estudado desde que a evolução tecnológica começou a expandir seu alcance, seus usuários e suas aplicações. As práticas de desenvolvimento evoluíram, necessitando de processos de desenvolvimento capazes de suportar a demanda de trabalho, abrindo caminho para o surgimento das metodologias ágeis.

Embora a utilização do processo de desenvolvimento ágil tenha reduzido o tempo gasto pelas equipes no ciclo de desenvolvimento de software, surgiram novos gargalos no processo que precisavam de novas soluções. O *DevOps*, termo que une os conceitos relacionados ao desenvolvimento (*dev*) com operação (*ops*), surgiu como um conjunto de práticas que focam em reduzir o tempo entre os *commits* e implantação em produção, por meio de uma integração de equipes e tarefas automatizadas (BASS, 2017).

Com a crescente necessidade de desenvolvimento contínuo (CD) e entrega e implantação contínua (CI) de *software*, novas arquiteturas de dados também emergiram, permitindo que o ambiente seja escalável, de fácil manutenção e reutilizável (PIANINI; NERI, 2021).

O método *DevDocOps* visa integrar documentação (*doc*) ao processo *DevOps*. A capacidade de suportar documentação de maneira contínua dentro de CI/CD pode trazer maior qualidade nos documentos gerados. Mediante referências diárias integradas às entregas, é possível obter características nos documentos gerados

como: precisão de informações, integridade de especificação e rastreabilidade da linha de tempo (RONG et al., 2019).

As ferramentas de automação de documentação existentes no mercado em sua maioria trabalham em cima de comentários de código criando guias de referência (SYNKO; PELESHCHYSHYN, 2020; PIANINI; NERI, 2021), porém existem outros artefatos de conhecimento técnico que podem ser obtidos e convertidos em documentos de alta qualidade, como classes, *tags*, pacotes, métodos, cenários e resultados de testes, erros e correções, entre outros (RONG et al., 2019; THEUNISSEN; van Heesch; AVGERIOU, 2022; PONISZEWSKA-MARAÑDA; ZIELISKI; MARAÑDA, 2020).

Na era atual, na qual o volume de dados cresce exponencialmente, a automação da documentação de dados emerge como uma solução crucial. Esta abordagem minimiza o trabalho manual necessário para atualizar a documentação existente, enfrentando desafios crescentes à medida que o volume de dados se expande. A automação facilita a concentração das equipes de dados em projetos de maior escala e garante a atualização constante da documentação em paralelo às mudanças nos dados. A documentação padronizada e automatizada não só melhora a confiança nos dados, mas também se adapta e escala com o crescimento dos dados e da empresa, eliminando a necessidade de revisões extensivas ou iterações constantes.

Embora a abordagem *DevDocOps* oriente a automatização da documentação e inclusão no *DevOps*, ela ainda não é capaz de superar todos os gargalos existentes. Devido ao fato do *DevDocOps* ter sido proposto em 2019 (RONG et al., 2019), sua aplicação prática ocorreu em ambiente particular, não sendo possível analisar a sua aplicação e resultados na comunidade de código aberto. A proposta apresentada por Rong et al. (2020) visa conceber um meio para a documentação ser automatizada, mas deixa a ser definido pelo desenvolvedor as informações, diretrizes e modelos de documentação que irão compor a documentação final entregue.

O *DevDocOps*, apesar de ser uma abordagem inovadora, ainda é considerada generalista e carece de um estabelecimento concreto de aplicação, especialmente no campo da engenharia de dados. Até o momento, não há padrões ou diretrizes precisas que definam como o fluxo de documentação deve ocorrer durante o desenvolvimento.

Essa falta de clareza e especificidade nos métodos de implementação do *DevDocOps* dificultam a sua aplicação prática e a padronização em projetos de engenharia de dados.

Além disso, considerando a engenharia de dados, a ausência de diretrizes consolidadas para a documentação no contexto do *DevDocOps* apresenta um desafio significativo. Sem um conjunto claro de práticas e padrões, é difícil para as equipes de engenharia de dados integrar efetivamente a documentação automatizada no ciclo de desenvolvimento. Esta lacuna na abordagem do *DevDocOps* destaca a necessidade de pesquisas e desenvolvimentos adicionais para estabelecer um método mais estruturado e eficaz para a documentação em projetos de engenharia de dados.

1.2 Objetivos

O objetivo geral desta pesquisa foi conceber uma abordagem para a realização do desenvolvimento e entrega de documentação contínua de engenharia de dados, no espírito de *DevDocOps*, identificando diretrizes concretas para otimizar a qualidade de documentação de arquitetura de dados.

Para alcançar o objetivo proposto foi realizada uma análise da literatura para verificar a existência de métodos já propostos para aplicação de metodologias ágeis na documentação técnica de engenharia de dados. Esta análise forneceu percepções acerca dos gargalos existentes na literatura e no mercado acerca do tema proposto, viabilizando a construção de um fluxo de documentação automatizada que possibilite a implementação prática do *DevDocOps*.

A partir da realização de um estudo experimental, conduzido em parceria com uma empresa colaboradora, foi possível conceber um conjunto de diretrizes capazes de orientar a aplicação da abordagem *DevDocOps* em projetos de Engenharia de Dados.

A partir do objetivo geral, os seguintes objetivos específicos foram definidos:

- Agilizar o desenvolvimento de documentação: Desenvolver e implementar um processo otimizado para acelerar a produção de documentação de objetos de

banco de dados desenvolvidos ou modificados, visando reduzir significativamente o tempo necessário para documentação.

- **Parametrizar a Qualidade da documentação:** Estabelecer padrões para assegurar a alta qualidade da documentação, incluindo detalhamento aprofundado dos objetos documentados e a correteude dos documentos produzidos.
- **Promover a Entrega Contínua de documentação:** Implementar um sistema de versionamento que garanta a sincronização entre a documentação e o código em produção em todas as *releases*, assegurando consistência e rastreabilidade.
- **Entrega de documentações:** Criar e manter um sistema acessível de distribuição de documentação, que facilite o acesso dos interessados às informações atualizadas do projeto.

1.3 Metodologia

Para atingir os objetivos, este trabalho foi organizado a partir de técnicas de pesquisa, como revisão bibliográfica, estudo exploratório e avaliação dos resultados.

Cada etapa será descrita a seguir:

- **[Etapa A] - Revisão Bibliográfica:** Estudo bibliográfico inicial para obter uma visão geral dos tópicos de *DevOps*, documentação de software em metodologias ágeis, documentação de software para engenharia de dados, automatização de documentação de engenharia de dados.
- **[Etapa B] - Entendimento do ambiente organizacional:** Foi realizada a investigação do ambiente e coleta de dados necessários para a concepção da abordagem em conjunto com uma empresa parceira.
- **[Etapa C] - Planejamento dos testes:** Foi realizada a definição das metas e objetivos, assim como uma elaboração de como as mesmas seriam alcançadas.
- **[Etapa D] - Investigação de Técnicas de Automatização:** Para entender como as documentações poderiam ser automatizadas, foram realizadas

pesquisas sobre documentação ágil, que levaram ao desenvolvimento de *scripts* de documentação *in loco*.

- **[Etapa E] - Construção e configuração do fluxo DevDocOps:** Para a aplicação prática da abordagem DevDocOps, foi realizada a elaboração do processo, escolha da ferramenta e parametrização do ambiente.
- **[Etapa F] - Testes e Avaliação:** Foi realizada a aplicação das diretrizes propostas no ambiente organizacional, com o apoio de uma equipe de voluntários. Os resultados obtidos por meio de questionários e entrevistas foram utilizados para a extração da avaliação dos métodos propostos.

1.4 Contribuições

A principal contribuição deste trabalho são os resultados do estudo experimental, realizado em parceria com uma empresa, que evidenciam os benefícios da abordagem *DevDocOps* para o desenvolvimento ágil. Indícios obtidos a partir do *feedback* fornecido pelos participantes do estudo experimental demonstram que, com a abordagem, é possível obter maior agilidade e qualidade na produção da documentação de software. Também foram observados benefícios oriundos da constante atualização da documentação.

Outra contribuição importante foram as diretrizes de documentação que foram levantadas nesta pesquisa. As lições aprendidas no processo proposto, tais como o passo-a-passo a ser realizado pelas diferentes partes interessadas, constituem dicas importantes para outros pesquisadores e praticantes interessados em adotar práticas *DevDocOps*. Entre essas lições, destaca-se a importância da participação integrada de toda a equipe, incluindo ações a serem realizadas pelo líder técnico e não apenas a equipe de desenvolvimento, a importância da sistematização do processo, incluindo etapas obrigatórias, entre outras diretrizes que, neste estudo, se mostraram benéficas.

Além disso, pode-se citar como contribuição os resultados do estudo bibliográfico. Esta é uma área recente, com poucos trabalhos publicados, e poucas ferramentas existentes que auxiliam na incorporação da documentação ao processo

DevOps. Contribuições na forma de revisão bibliográfica e comparação teórica podem ajudar futuras pesquisas sobre o assunto.

1.5 Organização do Trabalho

O restante deste trabalho está organizado em 4 capítulos principais, sendo eles: fundamentos e trabalhos relacionados; estudo experimental; diretrizes para a documentação de engenharia de dados em uma abordagem *DevDocOps*.

- **Capítulo 2 - Fundamentos e Trabalhos Relacionados:** apresenta a definição dos tópicos de pesquisa que permeiam o contexto proposto, como CI/CD, *DevOps*, Engenharia de Dados, Documentação Técnica de software e de dados, *DevDocOps* e Sap Hana. Além das definições serão apresentados os trabalhos relacionados a estes temas a fim de apresentar os estudos recentes dos tópicos relacionados a este trabalho.
- **Capítulo 3 - Estudo Experimental:** apresenta informações sobre o planejamento e aplicação do estudo realizado para aplicar em um ambiente empresarial controlado as diretrizes de documentação de dados dentro da abordagem *DevDocOps* de forma prática.
- **Capítulo 4 - Diretrizes para a Documentação de Engenharia de Dados em uma abordagem *DevDocOps*:** Apresenta a definição das diretrizes de aplicação da abordagem.
- **Capítulo 5 - Conclusão:** Apresenta as principais descobertas obtidas no projeto, as implicações práticas dos resultados do estudo experimental, as recomendações para futuras pesquisas e a conclusão final do trabalho.

2 Fundamentos e Trabalhos Relacionados

Neste capítulo, na Seção 2.1 são apresentados os principais conceitos que fundamentam este estudo. Na Seção 2.2 são apresentados os trabalhos que retratam relação com os temas desta pesquisa, derivados de um estudo da literatura, que teve por objetivo identificar os estudos que embasam a realização deste projeto.

Este trabalho tem como fundamento os conceitos relacionados à integração contínua (CI) e desenvolvimento contínuo (CD), engenharia de dados, documentação técnica de software e dados, e *DevDocOps*. As seções seguintes apresentam os fundamentos de cada conceito, contudo a intersecção de todos os temas discutidos ainda não foi pesquisada na literatura existente.

2.1 Fundamentos

2.1.1 Integração Contínua (CI) e Desenvolvimento Contínuo(CD)

A partir da criação do Manifesto Ágil¹ em 2001, os impulsos por pesquisa e definições de novos métodos de desenvolvimento cresceram em torno de suas diretrizes propostas, dando surgimento a métodos e abordagens de desenvolvimento de software ágil (LEITE et al., 2019; PONISZEWSKA-MARAÑDA; ZIELISKI; MARAÑDA, 2020).

Segundo Gokarna e Singh (2020), embora a utilização do processo de desenvolvimento ágil tenha reduzido o tempo gasto pelas equipes no ciclo de desenvolvimento de software, surgiram novos gargalos no processo que precisam ser solucionados, como comunicação, integração contínua (CI) e automação.

O processo de integração contínua de software é definido pela utilização de um *pipeline* de implantação automatizado, em que qualquer versão desenvolvida passe por etapas de testes automatizados e compilações, permitindo que desen-

¹ <https://agilemanifesto.org/iso/ptbr/manifesto.html>

volvedores enviem de forma mais rápida e segura, ao ambiente de produção, as atualizações de aplicações (LEITE et al., 2019; VADAVALASA, 2020).

A partir da concepção de mais iterações propostas no desenvolvimento ágil de software, o desenvolvimento e sua gestão foi particionado em pequenos elementos de projeto (UNDERWOOD, 2023), para que esses fragmentos possam ser desempenhados de forma mais rápida (BERHOUMA, 2020), definindo o processo que chamamos de desenvolvimento contínuo (CD).

Existe uma aceitação considerável de que as características do software - com metadados associados - devem ser adicionadas em fases meticulosamente mais pequenas, ou *sprints*, à medida que se constrói software complexo. Este procedimento também é seguido para as atividades de atualização e manutenção - mesmo quando os dados e metadados já foram previamente especificados, concebidos, recolhidos e consumidos em produção. (UNDERWOOD, 2023)

Para o funcionamento do processo CI/CD, é necessário que o código fonte resida com base em uma configuração de gestão de código, permitindo a integração das etapas de desenvolvimento e entrega, reduzindo o tempo de interação com o *deploy* e minimizando possíveis erros manuais (VADAVALASA, 2020).

2.1.2 DevOps

O *DevOps* trata-se de uma abordagem com foco principal em integração das equipes de desenvolvimento (*dev*) e operações (*ops*) por meio da construção de *pipelines* de automatização de processos de CI/CD (LEITE et al., 2019; YARLAGADDA, 2021; MUÑOZ; RODRÍGUEZ, 2021). A partir da transformação promovida pela aplicação da abordagem, é possível que os desenvolvedores realizem cruzamento de códigos, testes de unidade e compilação constante, permitindo não só entregas de códigos atualizados com alta frequência, mas também detecção de erros e resoluções mais rápidas de problemas (YARLAGADDA, 2021).

A compreensão do *DevOps* é difusa, conforme exposto por Muñoz e Rodríguez (2021). Os autores definem que *DevOps* “é baseado em diferentes perspectivas dependendo da fonte consultada, o que dificulta o entendimento e implementação

do *DevOps* como apenas implementação de práticas de automação”. Para compreender o *DevOps*, é necessário embasar-se no Manifesto Ágil, no qual a premissa “Indivíduos e interações mais do que processos e ferramentas” reflete-se na colaboração humana dividida em silos proposta pela abordagem, assim como, seu quarto princípio, baseia a proposta de integração de atividades e práticas na automação de processos de desenvolvimento de software, permitindo que os desenvolvedores possam construir, testar e liberar entregas de forma rápida e confiável.

Um processo *DevOps* é normalmente composto por diferentes etapas e fases, que podem ser periodicamente reiteradas para o correto desenvolvimento e funcionamento do software, de forma evolutiva e capaz de captar novos requisitos, funcionalidades e comportamentos decorrentes da operação. (CAPIZZI; DISTEFANO; MAZZARA, 2020)

Conforme o processo caracterizado por Atwal (2020), uma vez que o produto submetido ao *pipeline* satisfaz os requisitos de testes parametrizados, o mesmo passa a estar pronto para ser implantado em produção, sendo processado por meio de uma infraestrutura e configuração de orquestração que esteja em uma ferramenta de gestão da configuração, juntamente com o código a ser versionado.

O foco no fluxo contínuo a partir de práticas ágeis deve melhorar os resultados a curto prazo, no qual a implantação parcial de fragmentos reduz o risco operacional da implantação produtiva, permitindo que mais dados sejam coletados durante o processo automatizado (ATWAL, 2020; VADAVALASA, 2020).

Por se tratar de uma filosofia abrangente, o *DevOps* permitiu que seus princípios fossem especializados em domínios específicos de software, como por exemplo, contextos de inteligência artificial (*AI Ops*), *machine learning* (*MLOps*), e até mesmo gestão e engenharia de dados (*DataOps*) (CAPIZZI; DISTEFANO; MAZZARA, 2020). O manifesto *DataOps*² propõe que, além das equipes serem capazes de promover sua auto-organização e gestão, o código gerado por elas impacta diretamente na tomada de decisão sobre os dados desde seu acesso, modelagem e visualização.

² <https://dataopsmanifesto.org/pt-pt/>

2.1.3 Engenharia de Dados

A partir da ocorrência da Primeira Conferência Internacional de Engenharia de Dados Computacionais (COMPDEC), ocorrida em 1984, a engenharia de dados passou a ser vista formalmente como um conjunto de atividades que envolvem: o projeto de bancos de dados lógicos e físicos; metodologias de gerenciamento de dados; arquiteturas de computador para bancos de dados de conhecimento; tecnologia, implementação e operação para gerenciamento de dados; e ferramentas especializadas (RAMAMOORTHY et al., 1984). Nesta mesma conferência, Zimmerman (1984) defendeu que, embora as análises de sistemas estruturados fossem assertivas em relação a projetos comerciais, que hoje pode ser compreendida como uma área da engenharia de software, já existiam gargalos em relação à concepção de bancos de dados, que eram desenvolvidos e aplicados de forma genérica. Seu estudo propôs então que, para o desenvolvimento assertivo de bancos de dados, deveria ser aplicada uma sequência de passos que compreendesse o estudo dos dados da organização, a decomposição das funções de negócio, o tratamento de armazenamento do fluxo de dados, além de realizar a união entre as análises estruturas de sistemas que já existiam com novas análises que focassem em dados.

Conforme exposto por Reis e Housley (2023) atualmente, embora existam diversas técnicas de desenvolvimento de banco e dados e múltiplas ferramentas, o cenário literário ainda carece de uma definição formal de aplicações de engenharia de dados de forma coesa, não existindo padrões para a sua aplicação.

Engenharia de dados é o desenvolvimento, implementação e manutenção de sistemas e processos que recebem dados brutos e produzem informações consistentes e de alta qualidade que suportam casos de uso *downstream*, como análise de dados e *machine learning*. A engenharia de dados é a combinação de segurança, gerenciamento de dados, *DataOps*, arquitetura de dados, orquestração e engenharia de software. Um engenheiro de dados gerencia o ciclo de vida da engenharia de dados, começando com a obtenção de dados dos sistemas de origem e terminando com o fornecimento de dados para casos de uso, como análise ou aprendizado de máquina. (REIS; HOUSLEY, 2023)

2.1.4 Documentação Técnica de software e Dados

Devido ao Manifesto Ágil propor “*software* em desenvolvimento mais que documentação abrangente”, a quantidade de documentação técnica de software desenvolvida é baixa (THEUNISSEN; van Heesch; AVGERIOU, 2022). Embora o *DevOps* promova a aceleração de entrega de recursos de código, a documentação de suporte correspondente se torna uma preocupação na maioria dos projetos que o adotam (RONG et al., 2019). No Desenvolvimento Contínuo de *Software* (CSD), é comum que os desenvolvedores não mantenham a documentação em sincronia com o real desenvolvimento do projeto (THEUNISSEN; van Heesch; AVGERIOU, 2022).

Os estudos de Leite et al. (2019), Poniszewska-Marañda, Zieliski e Marañda (2020), Rong et al. (2019), Synko e Peleshchyn (2020), Theunissen (2020) demonstram que a documentação técnica de software visa alcançar os seguintes objetivos:

- Descrever os requisitos que já foram desenvolvidos e validados;
- Registrar conhecimento dos problemas e soluções encontrados durante o ciclo de vida;
- Garantir a integridade do conhecimento e informações acerca do software desenvolvido;
- Registrar e acompanhar resultados de testes acerca das versões desenvolvidas;
- Facilitar o reúso de código e manutenção de software.

Com a crescente demanda por software e análises mais assertivas para os negócios, uma enorme quantidade de dados está sendo gerada e utilizada para várias aplicações, trazendo um aumento no volume de dados em construção, deixando evidente a falta de consistência em sua representação, no formato e no seu significado (REIS; HOUSLEY, 2023).

A automação da documentação de dados minimiza o trabalho manual envolvido na atualização da documentação existente, porém se torna cada vez

mais desafiador à medida que o volume de dados aumenta, também aumentando a necessidade da equipe de dados para se concentrar em projetos de maior escala e garante que, à medida que os dados mudam, a documentação também seja atualizada.

A documentação padronizada aumenta a confiança nos dados, pois garante que está atualizada à medida que seus próprios dados são. A documentação de dados automatizada é capaz de escalar à medida que os dados e a empresa crescem, não havendo necessidade de uma grande revisão ou iterações constantes na documentação (MIZRAHI, 2023).

Em um ambiente de desenvolvimento ágil é comum a geração de artefatos de conhecimento técnico em diversas etapas do ciclo de desenvolvimento de software. Esses artefatos podem ser obtidos e convertidos em documentos de alta qualidade, como classes, *tags* - marcadores de código fonte, pacotes, métodos, cenários e resultados de testes, erros e correções, entre outros (RONG et al., 2019; THEUNISSEN; van Heesch; AVGERIOU, 2022; PONISZEWSKA-MARAÑDA; ZIELISKI; MARAÑDA, 2020; TANG et al., 2021).

Com a expansão de aplicativos e tomadas de decisões orientadas por dados na última década, os conjuntos de dados tornaram-se um meio de registrar e acessar informações de diversos escopos e domínios, tornando uma prática comum a inclusão de descrições de texto em cada coluna dos conjuntos de dados desenvolvidos na forma de um dicionário de dados (RASHID et al., 2020).

Os dicionários de dados são úteis para muitas tarefas de gerenciamento de dados, inclusive para auxiliar os usuários nos processos de conversão de dados, testar a geração de dados, validar dados e armazenar critérios de uso de dados. Ao armazenar dados em um sistema que adere à estrutura de um determinado dicionário de dados, esse documento pode ser usado para ajudar na validação ao inserir novos dados no sistema ou ao atualizar os dados existentes. (RASHID et al., 2020)

2.1.5 *Dev Documentation Ops*

A abordagem *DevDocOps* concentra-se em promover a documentação contínua em um projeto de software. A proposta de Rong et al. (2019) estabelece a

realização de automatização da documentação, de forma complementar à metodologia DevOps, tornando a equipe de desenvolvimento responsável por realizar a definição e configuração de parâmetros de informações a serem documentadas, criação do *template* do documento e realização de ajustes no documento antes da entrega final.

O método *DevDocOps* visa integrar documentação ao processo do desenvolvedor no ciclo de vida de desenvolvimento de software, também podendo ser aplicado à Engenharia de Dados. A capacidade de suportar documentação de maneira contínua dentro de CI/CD pode trazer maior qualidade nos documentos gerados. A partir de referências diárias integradas às entregas, é possível obter características nos documentos gerados como: precisão de informações, integridade de especificação e rastreabilidade da linha de tempo (RONG et al., 2019).

2.1.6 Modelos de dados desenvolvidos no SAP HANA

O SAP HANA (*High-Performance Analytic Appliance*) é uma plataforma de banco de dados em memória desenvolvida pela SAP SE. Essa tecnologia envolve banco de dados, processamento de dados em tempo real e capacidades avançadas de análise em uma única ferramenta, conforme descrito por Załęski (2021), Färber et al. (2012), suas principais características e benefícios são:

- *In-Memory Computing*, nos quais os dados são armazenados na memória principal do servidor, proporcionando acesso ultra rápido aos dados, acelerando significativamente o processamento e a análise;
- *UI* unificada para as ferramentas de modelagem BW/HANA³;
- Capacidade de processamento de dados em tempo real;
- Integra funcionalidades avançadas de ETL (Extração, Transformação e Carga) e permite a integração de dados de diversas fontes, incluindo sistemas SAP e não SAP;

³ <https://www.sap.com/brazil/products/technology-platform/hana/what-is-sap-hana.html>

- Por se tratar de um banco de dados relacional completo, ele suporta operações tradicionais de banco de dados, como consultas SQL, e oferece uma arquitetura colunar para otimizar consultas analíticas;
- Oferece suporte a análises avançadas, como processamento de linguagem natural, aprendizado de máquina e algoritmos preditivos;
- Escalabilidade de dimensionamento.

Um dos principais objetos de modelagem de dados existentes no SAP HANA são as *Calculation Views*. Modeladas a partir de uma interface gráfica, são utilizadas para modelar regras e lógicas de negócio, e retornar conjuntos de dados como saída. Devido às *Calculation Views* serem artefatos de bancos de dados estruturados, capazes de realizar cálculos complexos e aplicações de regras de negócio de forma massiva, a existência de documentação compreensiva e detalhada assume uma importância particular. O trabalho de [Załęski \(2021\)](#) apresenta um detalhado descritivo das *Calculation Views*, fornecendo uma análise abrangente de suas funções e explorando os diferentes tipos disponíveis. A explanação abrange tanto as funcionalidades gerais dessas visualizações quanto suas aplicações específicas, oferecendo uma visão aprofundada da diversidade e utilidade das *Calculation Views* no contexto do SAP HANA:

- *Calculation Views* Base: Este tipo de *Calculation Views* é utilizado para combinar dados de tabelas base e executar cálculos simples. Ele oferece funcionalidades básicas de agregação e projeção, sendo ideal para operações mais simples.
- *Calculation Views* com União: Esse tipo de *Calculation Views* permite unir dados de diferentes fontes, como tabelas ou outros *Calculation Views*. Ele é útil para integrar informações de diversas fontes e fornecer uma visão unificada.
- *Calculation Views* Hierárquico: O *Calculation Views* Hierárquico é projetado para representar hierarquias em dados, oferecendo suporte à visualização de

dados em uma estrutura hierárquica, como organogramas ou estruturas de categorias.

Quanto à estrutura interna de uma *Calculation View* no SAP HANA, elas são geralmente definidas em linguagem SQL e representadas por meio de um arquivo XML. O script SQL define as operações de modelagem e os cálculos a serem aplicados, enquanto o arquivo XML descreve a estrutura hierárquica e os relacionamentos entre os diferentes elementos (FÄRBER et al., 2012).

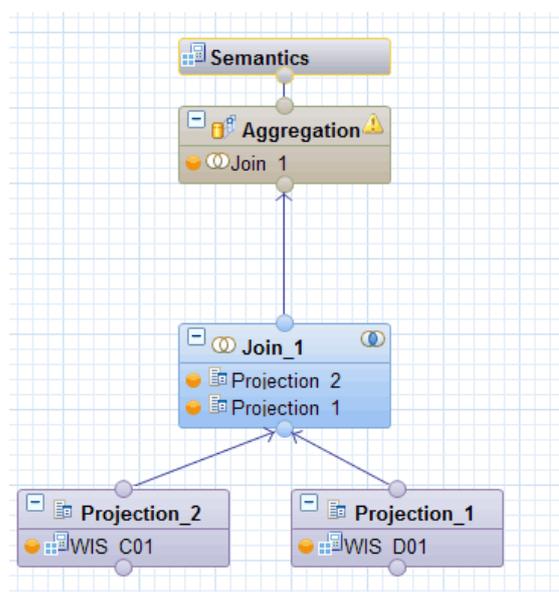


Figura 1 – Exemplo simplificado de estrutura de Calculation View
<https://blogs.sap.com/2019/12/02/sap-hana-calculation-views/>

Dentro de uma *Calculation View*, pode-se encontrar:

- Tabelas de Entrada (*Source Tables*): Representam as fontes de dados originais.
- Projeções (*Projections*): Definem a estrutura de saída da *Calculation View*.
- Junções (*Joins*): Permitem combinar dados de diferentes fontes.
- Agregações (*Aggregations*): Realizam operações de agregação nos dados.

- Colunas Calculadas (*Calculated Columns*): Permitem a criação de novas colunas com base em cálculos específicos.

Os dados resultantes da utilização de *Calculation Views* não apenas desempenham um papel crucial na modelagem de dados eficiente, mas também se revelam fundamentais para a extração de *insights* estratégicos em tempo real dentro da plataforma. À medida que as organizações continuam a explorar as potencialidades do SAP HANA, as *Calculation Views* emergem como peças-chave na busca por soluções analíticas avançadas e na transformação eficaz de dados brutos em conhecimento.

2.2 Trabalhos Relacionados

A pesquisa realizada por [Theunissen \(2020\)](#) aborda o equilíbrio entre documentação e comunicação nas práticas modernas de desenvolvimento de software como *Lean*, *Agile* e *DevOps*, coletivamente denominadas Desenvolvimento de Software Contínuo (CSD). Ele destaca os desafios de manter uma comunicação eficaz com documentação mínima, porém suficiente, nesses ambientes. O artigo propõe um quadro de documentação adaptado ao CSD, estabelecendo condições necessárias e suficientes para uma comunicação eficaz com a documentação justa, considerando "condições necessárias" os requisitos mínimos para se documentar, e "condições suficientes" a ocorrência de fato da comunicação com base na documentação provida. Sugere-se mais investigações sobre como diferentes tipos de informações e razões para documentação se correlacionam em diversas indústrias no CSD e como organizar informações dispersas em documentação compreensível.

O pesquisador ([AGHAJANI et al., 2019](#)) direcionou o seu estudo com o intuito de investigar as necessidades dos profissionais de software em relação à documentação automática de alta qualidade. Ao se deparar com a limitação em comum de outros pesquisadores como sendo o baixo número e falta de diversidade dos profissionais entrevistados e dos artefatos de documentação analisados, adotou uma abordagem qualitativa para mitigar esses problemas, analisando diferentes tipos de artefatos de diversas fontes de dados, como repositórios de software, e-

mails de desenvolvimento, discussões em fóruns de programação, problemas e *pull requests* relacionados à documentação de software. A pesquisa apresentada realizou o levantamento de diferentes ferramentas e abordagens para a documentação de software automatizada existentes na literatura, porém sua conclusão define que a documentação automatizada é ainda um gargalo, no qual uma documentação incorreta pode ter consequências graves imprevistas, que vão para além da perda de tempo a tentar reproduzir um exemplo de código errado.

Segundo [Rong et al. \(2019\)](#), uma forma de mitigar o problema de documentação técnica é por meio da incorporação do trabalho de documentação de software dentro da abordagem *DevOps*, nomeando esta abordagem de *DevDocOps*. Embora a abordagem *DevDocOps* oriente a automatização da documentação e inclusão no *DevOps*, ela ainda não é capaz de superar todos os gargalos existentes. Uma das primeiras citações ao termo *DevDocOps* ocorreu em 2019 [Rong et al. \(2019\)](#). Sua aplicação prática ocorreu em ambiente particular, não sendo possível analisar a sua aplicação e resultados na comunidade de código aberto. A proposta apresentada por [Rong et al. \(2020\)](#) visa conceber um meio da documentação a ser automatizada, mas deixa a ser definido pelo desenvolvedor as informações, diretrizes e modelos de documentação que irão compor a documentação final entregue.

As ferramentas de automação de documentação existentes no mercado, em sua maioria, trabalham em cima de comentários de código criando guias de referência ([SYNKO; PELESHCHYSHYN, 2020](#); [SONG et al., 2020](#)), porém existem outros artefatos de conhecimento técnico que podem ser obtidos e convertidos em documentos de alta qualidade, como classes, tags, pacotes, métodos, cenários e resultados de testes, erros e correções, entre outros ([RONG et al., 2019](#); [THEUNISSEN; van Heesch; AVGERIOU, 2022](#); [PONISZEWSKA-MARAÑDA; ZIELISKI; MARAÑDA, 2020](#)) .

Ao investigar o cenário atual do mercado em relação às ferramentas existentes destinadas à documentação automatizada de engenharia de dados, foi realizada uma análise para explorar as capacidades, limitações e aplicações práticas dessas ferramentas, oferecendo um panorama abrangente de como a automação está redefinindo as práticas de documentação em diversos setores. Ao comparar ferramentas de

automação de documentação de dados com a abordagem *DevDocOps*, é importante primeiro entender o que cada um oferece e onde podem haver lacunas ou limitações.

- **Secoda**⁴

Esta ferramenta facilita a tarefa de automatizar a documentação de dados. Ela usa metadados para registrar automaticamente informações como quando uma tabela ou banco de dados foi documentado pela última vez, quem documentou, consultas executadas contra essa tabela, descrições de tabelas e colunas, linhagem de colunas e tabelas, perfis de colunas e métricas. Secoda também permite a documentação automática das interações dos usuários com os dados, o que ajuda a entender quais recursos são irrelevantes ou obsoletos.

Limitações: Embora ofereça automação e atualização contínua, pode não integrar totalmente a documentação no ciclo de vida de desenvolvimento de dados, uma vez que o foco principal está na automação da documentação, e não na integração profunda com práticas de desenvolvimento e operações

- **Apache Spark e Apache Airflow**

*Spark*⁵ é um motor analítico unificado de código aberto para processamento de dados em larga escala, enquanto *Airflow*⁶ é uma plataforma de gerenciamento de fluxo de trabalho de código aberto. Ambas são ferramentas essenciais no mundo da engenharia de dados para processamento e gerenciamento de grandes conjuntos de dados.

Limitações: Estas ferramentas são mais focadas no processamento e gerenciamento de dados do que na integração de documentação no ciclo de vida de desenvolvimento.

Enquanto as ferramentas de documentação automatizada fornecem soluções eficientes para a gestão de dados e documentação, elas podem não atender totalmente aos requisitos do *DevDocOps* em termos de integração completa de documentação no ciclo de vida de desenvolvimento e operações. A abordagem

⁴ <https://www.secoda.co/data-catalog>

⁵ <https://spark.apache.org/>

⁶ <https://airflow.apache.org/>

DevDocOps requer uma integração mais profunda e contínua da documentação, algo que pode ser desafiador para ferramentas projetadas principalmente para automação de tarefas específicas de dados.

Ao analisar o uso do SAP HANA como Sistema Gerenciador de Banco de Dados (SGBD), observa-se a ausência de ferramentas externas ao SAP que sejam capazes de documentar seus objetos, e mesmo com as ferramentas internas, a padronização da documentação é um desafio. Uma estratégia usual é empregar a funcionalidade de Auto-Documentação do SAP HANA Studio⁷, esta opção permite a criação de documentações em formato PDF, contudo, essas documentações automáticas frequentemente não apresentam todas as informações necessárias de forma útil ou adaptada para elaborar documentos específicos, como um documento de mapeamento.

⁷ <https://blogs.sap.com/2012/03/31/auto-documentation-functionality-in-hana-studio/>

The screenshot shows a PDF report generated by SAP HANA Studio. The report is titled 'Analytic View Report' and provides detailed information about an analytic view named 'AN_MK_NEW_SALES_1'. The report is organized into several sections:

- General:** A table with columns 'Name' and 'Value'. It lists the name, status (active), owner (MURALO), package (muralo-new-test-package), description (AN_MK_NEW_SALES_1), and creation date (2012-03-14 12:42:26.414).
- Private Attributes:** A table with columns 'Name', 'Description', 'Schema (Active Column)', and 'Data Type'. It lists attributes like MANDT_ID, PRODUCT_ID, CUSTOMER_ID_1, YEAR, and SOLD_DATE with their respective data types.
- Calculated Attributes:** A table with columns 'Name', 'Description', 'Formula', and 'Datatype'. It currently shows 'None'.
- Attribute Views:** A table with columns 'Name', 'Package', and 'Description'. It lists attribute views AT_MK_CURR, AT_MK_NEW_CUSTOMER, and AT_MK_NEW_PRODUCT.
- Measures:** A table with columns 'Name', 'Description', 'Schema, Table, Column', and 'Data Type'. It lists measures NET_SALES and BILLED_QUANTITY with their data types.
- Calculated Measures:** A table with columns 'Name', 'Description', 'Formula', and 'Data Type'. It currently shows 'None'.

The report footer indicates it was generated on 31.03.2017 07:55:17 and is page 1 of 2.

Figura 2 – Documento automático gerado no caso de detalhes do modelo
<https://community.sap.com/t5/technology-blogs-by-sap/auto-documentation-functionality-in-hana-studio/ba-p/13209456>

Outra alternativa é utilizar um método que envolve a exportação dos dados de *Calculation View* em formato XML do SAP HANA Studio e depois usar uma macro baseada no Excel para gerar um documento de mapeamento detalhado ⁸.

Entretanto, a documentação produzida no SAP HANA demanda um esforço considerável para a adequação aos padrões desejados e para o acréscimo de informações específicas necessárias para a entrega desenvolvida. Adicionalmente, essa abordagem de documentação não possui uma integração com as práticas de gestão ágil de *software*.

⁸ <https://blogs.sap.com/2018/04/17/tips-sap-hana-auto-documentation-excel-way/>

3 Estudo Experimental

3.1 Introdução

Esta pesquisa visou criar e avaliar uma abordagem para agilizar a produção e entrega contínua de documentação de engenharia de dados, alinhada com a filosofia *DevDocOps*. O enfoque central concentra-se em estabelecer diretrizes que aprimorem a qualidade da documentação de engenharia de dados, por meio de um processo automatizado inserido em um *pipeline* de CI/CD.

A concepção da pesquisa experimental foi delineada por meio de etapas distintas, que serão descritas nos tópicos subsequentes. A etapa de planejamento teve como propósito investigar e definir o escopo do projeto, o ambiente técnico de aplicação e o plano de testes. Na fase de execução, concentrou-se na parametrização e desenvolvimento do ambiente ferramental necessário para a implementação eficaz da filosofia *DevDocOps*. Os resultados obtidos, provenientes de testes conduzidos em ambiente controlado, serviram como validação da diretriz proposta, fornecendo percepções cruciais para sua análise e posterior aprimoramento contínuo.

3.2 Planejamento

A concepção do planejamento foi delineada por meio de três etapas, descritas a seguir:

Na etapa inicial do planejamento da pesquisa, denominada Etapa A, foi realizado um levantamento bibliográfico abrangente, com o objetivo de definir o escopo inicial do projeto, com foco nas práticas de documentação dentro da metodologia ágil. A exploração do tema abordou a compreensão da abordagem *DevDocOps*, suas aplicações e áreas de melhoria, além de avaliar a viabilidade de aplicação em domínios específicos da Engenharia de Software. Como parte integrante desse processo, a área de domínio foi delimitada, sendo especificamente identificada como Engenharia de Dados.

A segunda etapa, denominada Etapa B, teve como foco o planejamento da aplicação técnica, no qual foi realizada a busca e seleção de uma empresa parceira capaz de fornecer percepções e insumos sobre seus processos atuais, além de colaborar na aplicação de testes experimentais relacionados à abordagem proposta neste trabalho. Esse processo envolveu a realização de reuniões de ideação, apresentação de propostas e o planejamento detalhado dos testes a serem conduzidos. As informações coletadas concentraram-se na definição e compreensão de documentações técnicas e suas aplicações, na análise de padrões de documentação no desenvolvimento ágil e na delimitação dos objetos de banco de dados que seriam abordados na documentação automatizada proposta por este projeto.

Na terceira etapa, denominada Etapa C, voltada para o planejamento detalhado dos parâmetros de teste, foi realizada a definição da plataforma de *DevOps* e das ferramentas que seriam empregadas para conceber e aplicar as diretrizes propostas nesta pesquisa. Um ponto essencial desse planejamento envolveu a adoção da abordagem GQM (*Goal - Question - Metric* ou Meta - Perguntas - Métricas), com o intuito de estabelecer metas específicas e objetivos claros que esta pesquisa visou alcançar. A metodologia GQM desempenha um papel crucial ao fornecer uma estrutura estratégica para a definição e mensuração de metas, identificando as métricas relevantes e formulando perguntas específicas que orientam o processo de coleta de dados.

Dentro desse contexto, foi elaborado um plano de testes abrangente que seria executado no ambiente organizacional, contando com uma equipe pré-definida e alinhada com a empresa parceira. Esse plano foi projetado para validar a aplicação das diretrizes propostas, garantindo uma abordagem sistemática e abrangente na avaliação do impacto das mudanças introduzidas pela pesquisa. A fase de planejamento estabeleceu as bases para a execução dos testes, assegurando uma avaliação robusta e alinhada com os objetivos previamente definidos.

3.2.1 Etapa A - Revisão Bibliográfica

Para realizar a definição do tema de abrangimento da pesquisa, foi analisada a pesquisa de [Theunissen \(2020\)](#), [Theunissen, Hoppenbrouwers e Overbeek \(2022\)](#), que identificou e caracterizou a existência de um gargalo na documentação técnica

de projetos de software desenvolvidos e estruturados por meio de metodologias ágeis.

A partir da pesquisa realizada por [Rong et al. \(2019\)](#), pode-se conceber uma abordagem de documentação automatizada de *software*, dentro do ciclo de vida *DevOps*, utilizando métodos de CI/CD. Partindo da análise da premissa proposta pelo *DevDocOps*, foi realizado um estudo sobre a possibilidade de aplicação da abordagem em nichos específicos de desenvolvimento de *software* que careciam de uma definição concreta de padrões de documentação, estabelecendo a definição do escopo da pesquisa.

Com base na observação das práticas empregadas na área de Engenharia de dados, foi possível constatar a existência de uma diversidade de abordagens e metodologias existentes no mercado quanto à gestão e criação de documentos técnicos dos projetos desenvolvidos por empresas do ramo.

3.2.2 Etapa B - Análise e Entendimento do ambiente organizacional no domínio da pesquisa

Com a finalidade de obter informações práticas acerca das demandas e lacunas existentes na área em estudo, estabeleceu-se uma parceria com uma empresa que atuasse como fonte de dados práticos para o projeto em questão. A empresa selecionada como parceira é uma entidade de significativo impacto no mercado nacional de tecnologia, destacando-se no cenário de consultoria e desenvolvimento de software no território brasileiro.

O processo inicial envolveu a condução de reuniões destinadas à apresentação da proposta e à definição de uma metodologia de pesquisa que assegurasse a manutenção do anonimato da organização. Estas reuniões foram realizadas em colaboração com um gerente de engenharia de dados, figura-chave responsável pelo pilar de projetos de dados envolvendo as tecnologias SAP na empresa.

A segunda etapa compreendeu reuniões de *brainstorming* individuais com dois Gerentes de Engenharia de Dados, bem como com um Gerente Sênior da área, visando a ideação e avaliação dos objetivos do projeto e possíveis aplicações práticas.

Por meio da seleção da empresa, que disponibilizou um consultor de engenharia de dados para acompanhar e apoiar voluntariamente o desenvolvimento e evolução do projeto, estabeleceram-se diálogos colaborativos. Essa interação culminou na decisão estratégica de utilizar o ambiente de negócios da empresa como cenário para a condução de análises e testes relevantes à pesquisa.

Essa abordagem prática proporcionou uma integração sinérgica entre a investigação acadêmica e as dinâmicas reais do contexto empresarial, enriquecendo substancialmente a aplicabilidade e relevância dos resultados obtidos.

A partir do acompanhamento e análise do processo de documentação em dois projetos distintos de desenvolvimento de engenharia de dados, foi possível mapear as discrepâncias e similaridades entre os grupos, mesmo que pertençam à mesma cadeia tecnológica. Ambas as equipes compartilham características comuns, como envolvimento de mais de cinco desenvolvedores atuantes, uma maturidade organizacional estabelecida (existindo por mais de seis meses) e experiências com trocas de participantes ao longo de sua existência. Esses aspectos adicionais fornecem um contexto mais abrangente para a compreensão das dinâmicas de cada equipe, suas práticas de documentação e impacto das informações geradas. Os dados colhidos são apresentados na Tabela 1:

Tabela 1 – Comparação de Artefatos entre Equipes.

Artefato de Observação	Projeto A	Projeto B
Desenvolvimento de Documentação Técnica	Sim	Sim
Modelo Padrão de Documentação Técnica	Sim	Não
Atualização constante das documentações construídas	Às Vezes	Às Vezes
Descritivo Técnico de Regras Desenvolvidas	Não	Não
Descritivo de Negócio de Regras Desenvolvidas	Sim	Sim
Documentação de <i>Calculation Views</i>	Sim	Não
Documento de Testes Unitários	Sim	Não
Dicionário de Dados	Sim	Sim
Modelo Padrão de dicionário de dados	Sim	Sim
Necessidade de recursos humanos focados em apenas documentar	Sim	Sim

Fonte: Elaborado pela autora.

Por meio da constatação das discrepâncias e similaridades encontradas dentro do ambiente organizacional entre as abordagens empregadas para a documentação dentro de projetos, os insumos colhidos revelaram-se como um catalisador

significativo para a complementação do escopo desta pesquisa, visando não apenas compreender as variáveis que impactam as práticas de documentação técnica de dados, mas também contribuir para a evolução e aprimoramento do campo da engenharia de dados.

Na condução desta pesquisa, a análise e descoberta dos tipos de documentações de engenharia de dados desenvolvidas pelas equipes foram realizadas na estrutura operacional da empresa. Este processo incluiu uma investigação aprofundada sobre os procedimentos cotidianos necessários para conceber tais documentações. Ao explorar *in loco*¹ as práticas adotadas, foi possível elucidar não apenas a diversidade de documentações, mas também os intrincados processos diários envolvidos em sua concepção. A variabilidade de documentações foi mapeada na Tabela 2:

Tabela 2 – Mapeamento de Artefatos de Documentação de Dados.

Tipo de Documento	Conteúdo	Arquivo
Dicionário de Dados	Extração do descritivo de campos, suas tipologias, e comentários que compõe uma Tabela.	Excel
Documento de Testes Unitários	Imagens dos resultados obtidos na execução de Tabelas ou <i>Calculation Views</i> desenvolvidas.	Word
Documentação de <i>Calculation Views</i>	Imagens capturadas de cada nó existente no fluxo da <i>Calculation Views</i>	Word
Descritivo de Negócio de Regras	Descritivo a nível de negócio de todas as regras e transformações que precisam ser desenvolvidas em uma determinada Tabela ou <i>Calculation Views</i> .	Word ou Excel

Fonte: Elaborado pela autora.

No contexto do ambiente organizacional, observou-se que, ao longo das entregas faseadas, seguindo tanto a metodologia do projeto ágil quanto a abordagem cascata, espera-se produzir documentos com o intuito de registrar tecnicamente os principais recursos ou funcionalidades implementadas. Estes documentos abarcam elementos como visual, *scripts* em nível moderado de descrição, modelagem e evidências do funcionamento dessas funcionalidades. Contudo, foi constatado que, no ambiente organizacional em questão, não há uma padronização rigorosa em relação

¹ *In loco* é uma expressão em latim, que significa “no lugar” ou “no próprio local” e é equivalente à expressão *in situ*.

à fase específica para elaboração ou ao formato a ser adotado para esses documentos, havendo variabilidade de acordo com a equipe, cliente e gestão envolvida em cada projeto de engenharia de dados.

Confrontando os resultados de pesquisa de [Leite et al. \(2019\)](#), [Poniszewska-Maraña, Zieliski e Maraña \(2020\)](#), [Rong et al. \(2019\)](#), [Synko e Peleshchyn \(2020\)](#), [Theunissen \(2020\)](#) com o ambiente organizacional, foi realizada a definição de especificação técnica como um documento detalhado e abrangente que descreve todos os procedimentos técnicos relacionados ao desenvolvimento de produtos de dados, contendo informações essenciais e detalhes do processo de desenvolvimento de objetos. A documentação em questão é caracterizada por sua natureza dinâmica, refletindo sua variabilidade mediante atualizações e modificações contínuas ao longo do ciclo do projeto ou desenvolvimento.

No âmbito da metodologia de desenvolvimento ágil, a pesquisa direcionada revelou a ausência de padrões uniformes ou predefinidos no que diz respeito à documentação de engenharia de dados. A análise aprofundada demonstrou que não foram identificadas diretrizes específicas ou normativas que determinassem a estrutura, o formato ou o conteúdo desse tipo de documentação dentro do contexto ágil. Esta constatação ressalta a natureza flexível e adaptativa dessa abordagem, deixando espaço para uma diversidade de práticas e estilos de documentação entre as equipes envolvidas em projetos ágeis de engenharia de dados.

Em conjunto com a empresa, a pesquisadora realizou uma análise das entregas efetuadas por uma equipe de desenvolvimento de engenharia de dados, composta por três desenvolvedores de diferentes níveis, sendo eles: um desenvolvedor júnior, um desenvolvedor sênior, e um líder de equipe, com o intuito de realizar a definição de objetos de interesse. Notou-se que certos elementos, desenvolvidos com maior frequência, demandavam uma documentação uniforme. Nesse contexto, deliberou-se pela escolha do Dicionário de Dados como um dos documentos de interesse para fazer parte do novo processo, amparando-se também na pesquisa de [Rashid et al. \(2020\)](#) que classifica os dicionários de dados como documentação essencial para projetos baseados em dados.

Com o propósito de fomentar a inovação, optou-se por incluir também outro componente recorrente nos desenvolvimentos da equipe, as *Calculation Views*, objetos de visualização gráfica existentes no banco de dados utilizado, o *SAP HANA*. A definição desse objeto pela pesquisadora foi definida considerando os fatores impactantes na documentação desse elemento no processo existente da empresa. No método ágil, é atribuída uma prioridade mais elevada às tarefas com elevado valor para o cliente, o que pode levar à não priorização de tarefas de preservação da qualidade, como a atualização da documentação (JAIN; SUMAN, 2021), esta circunstância pode se refletir na não sincronização da documentação com a última versão do componente, bem como na variabilidade da qualidade e padrão, a depender do nível de conhecimento técnico do desenvolvedor responsável pela documentação. Esse cenário ressalta a importância de uma gestão eficaz do processo de documentação, visando minimizar o desperdício de recursos, evitar retrabalho e garantir uma integração fluida e consistente do software desenvolvido.

Conforme discutido no Capítulo 2, as *Calculation Views* no SAP HANA são objetos fundamentais para a modelagem de dados e análise dos dados ingestados na plataforma. Elas são utilizadas para criar visualizações complexas e personalizadas de dados, permitindo a combinação de diferentes fontes de dados, aplicação de lógica de negócios e criação de cálculos avançados (FÄRBER et al., 2012).

3.2.3 Etapa C - Definição dos parâmetros de teste

A plataforma *Azure DevOps* foi escolhida como ferramenta para aplicação da abordagem *DevDocOps*. Tal escolha foi justificada por alguns fatores de aplicação necessários para o funcionamento do processo, independente do ambiente organizacional que o aplique. Primeiramente, a integração nativa e abrangente oferecida pelo *Azure DevOps* é capaz de centralizar em uma única plataforma as funcionalidades de gerenciamento de projeto, controle de versão, *pipelines* de CI/CD, acompanhamento de tarefas e *wikis*², tornando possível obter uma sinergia eficiente, permitindo uma gestão holística do ciclo de vida do desenvolvimento (ROSSBERG, 2019).

² A Wiki no Azure DevOps é uma ferramenta de colaboração e documentação integrada à plataforma Azure DevOps Services.

A capacidade do *Azure DevOps* em suportar práticas *DevOps*, como integração contínua, entrega contínua e automação de processos, alinha-se perfeitamente com os princípios fundamentais da abordagem *DevDocOps*. Isso proporciona uma base sólida para a implementação da automação na documentação, integração entre desenvolvimento e documentação, e entrega contínua de documentação atualizada.

Outro aspecto relevante foi a robustez e a escalabilidade da plataforma, tornando-a adequada para projetos de diversos tamanhos e complexidades. A capacidade de personalização e a variedade de extensões disponíveis também foram consideradas, permitindo adaptar a ferramenta conforme as necessidades específicas da pesquisa.

Além disso, a integração nativa com serviços em nuvem, como o *Azure Repos* para controle de versão e o *Azure Pipelines* para automação de pipelines CI/CD, proporcionam uma experiência integrada, simplificando a configuração e a administração de todo o fluxo de trabalho.

Visando aprimorar os processos, a pesquisadora realizou a definição de métricas com base na metodologia GQM (*Goal, Question, Metric*), que visa estabelecer e medir os objetivos de um projeto ou processo. Ele é estruturado em três elementos principais: metas (*Goals*), perguntas (*Questions*) e métricas (*Metrics*). Essa abordagem é amplamente utilizada em engenharia de software e gestão de projetos para fornecer uma estrutura clara na definição, compreensão e avaliação de metas e objetivos (SOLINGEN; BERGHOUT, 2001). A aplicação dessa abordagem sistemática permitiu selecionar objetivos específicos, formular questionamentos pertinentes e, conseqüentemente, estabelecer métricas apropriadas para avaliar o desempenho e compreender de maneira abrangente a eficácia contínua do processo de melhoria delineado.

As metas representam os objetivos gerais e aspirações do projeto ou processo. Elas são declarações abrangentes que indicam a direção desejada e o propósito do trabalho a ser realizado, estabelecendo o contexto e o propósito, fornecendo uma visão clara do que se espera alcançar.

Conforme ilustrado na Tabela 3, com base na definição das metas (Goals), foi possível realizar a delimitação do nível conceitual que suportam os objetivos de negócio abordados neste projeto.

Tabela 3 – Estabelecimento de Metas.

Meta	Objeto	Finalidade	Ponto de Vista
M1 - Agilidade	Tempo para a produção de documentação	Otimização do processo de documentar um objeto de banco de dados desenvolvido ou modificado	Desenvolvedor
M2 - Qualidade	Qualidade da Documentação	Padronização da documentação, detalhamento do objeto documentado e corretude do documento	Gerente de projeto, Desenvolvedor e Cliente
M3 - Entrega Contínua	<i>Release</i> ou versão produtiva	Documentação e Código em produção possuírem a mesma versão em todas as entregas	Desenvolvedores, Gerente de Projetos
M4 - Entrega de Documentação	Documentação escrita	Documentação chegar aos atores interessados	Desenvolvedores, Gerente de Projetos, Cliente

Fonte: Elaborado pela autora.

As perguntas são formulações específicas que buscam obter informações detalhadas sobre aspectos específicos relacionados às metas, ajudando a traduzir metas abstratas em questionamentos mais tangíveis e mensuráveis no contexto estudado. Essas perguntas foram estrategicamente divididas em duas áreas de domínio, direcionadas aos desenvolvedores e líderes envolvidos no processo. Essa abordagem possibilitou a formulação de um questionário abrangente, adaptado às perspectivas e responsabilidades específicas de cada grupo.

As perguntas também foram elaboradas considerando-se que os participantes (desenvolvedores e líderes) deveriam respondê-las depois de terem realizado tarefas de documentação, conforme será explicado mais adiante neste capítulo.

Dentro do domínio de perspectiva dos Desenvolvedores, foram estabelecidas as seguintes perguntas:

- Q1: Foi produzida ou atualizada documentação nessa estória?

- Q2: Quanto tempo foi gasto (horas) com a documentação, somente na geração automática e revisões?
- Q3: Se não tivesse sido utilizado o processo DevDocOps, você estimaria ter gastado quanto tempo?
- Q4: O objeto em desenvolvimento já possui documentação?
- Q5: Caso sua resposta tenha sido “sim” para a questão 4, responda: Você precisou consultar a documentação anterior deste objeto para realizar a tarefa?
- Q6: Caso sua resposta tenha sido “sim” para a questão 4, responda: A documentação anterior estava atualizada com a mesma versão do objeto em produção?
- Q7: Caso sua resposta tenha sido “sim” para a questão 4, responda: Foi fácil acessar ou localizar a documentação?
- Q8: Foi necessário desenvolver alguma documentação de forma manual? Se sim, descreva.
- Q9: A documentação precisou de ajustes após ter sido gerada? Se sim, quais?
- Q10: Com o novo processo, a documentação tornou-se obrigatória. Como você avalia os benefícios e problemas dessa entrega contínua de documentação?
- Q11: Quais os benefícios e dificuldades que você constatou sobre a forma de entrega (disponibilização, acesso e visualização) da documentação na plataforma?

A conexão entre as metas definidas para a perspectiva de Desenvolvedores e as perguntas formuladas é ilustrada na Figura 3:

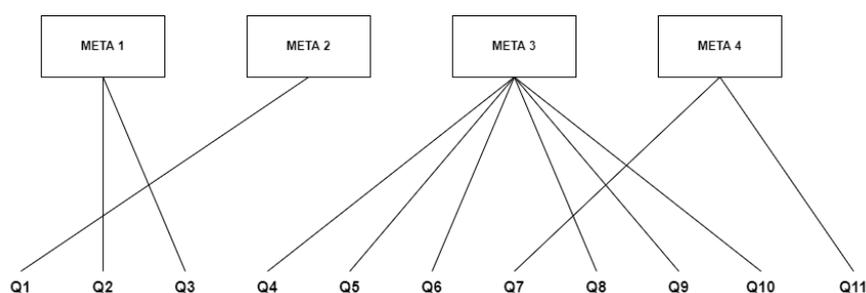


Figura 3 – Relacionamento entre as Metas e Perguntas de Pesquisa com os Desenvolvedores.

Fonte: Elaborado pela autora.

Dentro do domínio de perspectiva dos Líderes, foram estabelecidas as seguintes perguntas:

- Q1: A agilidade da documentação foi impactada positiva ou negativamente com a implantação desse processo?
- Q2: O que melhorou e o que piorou em relação à agilidade do desenvolvimento das estórias?
- Q3: A quantidade de documentação entregue na *sprint* aumentou.
- Q4: Em termos de padronização da documentação, houve melhoria?
- Q5: Em comparação ao processo de documentação anterior com o novo processo, a documentação entregue cumpre a necessidade técnica?
- Q6: Com o novo processo, a documentação tornou-se obrigatória. Como você avalia os benefícios e problemas dessa entrega contínua de documentação?
- Q7: Quais os benefícios e dificuldades que você constatou sobre a forma de entrega (disponibilização, acesso e visualização) da documentação na plataforma ?
- Q8: A documentação está sendo continuamente entregue? O que poderia melhorar?

A conexão entre as metas definidas para a perspectiva de Líderes e as perguntas formuladas é ilustrada na Figura 3.2.3:

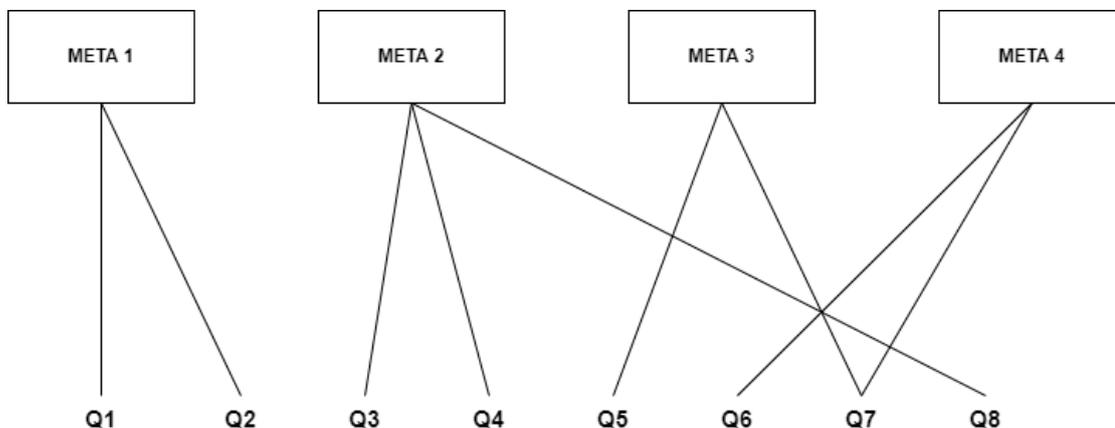


Figura 4 – Relacionamento entre as Metas e Perguntas de Pesquisa com os Líderes

Fonte: Elaborado pela autora.

As perguntas elaboradas no questionário são diretamente relacionadas às metas estabelecidas na Tabela 3, conforme representados pelas Figuras 3 e 3.2.3 representando uma abordagem sistemática para traduzir metas abstratas em questionamentos mais específicos e mensuráveis.

Além das questões definidas no GQM, houve a necessidade de incluir perguntas adicionais para obter informações mais específicas do desenvolvedor que estava respondendo, assim como para capturar detalhes sobre o contexto e a data da execução do teste. Essa abordagem visa enriquecer a coleta de dados, proporcionando análises mais detalhadas e contextuais sobre as percepções e experiências dos usuários durante o processo de aplicação.

A definição clara das metas e sua abrangência foi necessária para a concepção das perguntas (*Questions*) aplicadas ao nível operacional, a equipe envolvida no projeto, para clarificar e aperfeiçoar os objetivos, e captar as variações de compreensão em cada domínio, permitindo a definição das métricas (*Metrics*) que fornecerão respostas para as perguntas, descrevendo o nível qualitativo.

Para os desenvolvedores, a adoção das diretrizes propostas no dia a dia foi captada com base em *feedback* contínuo, possibilitando a extração de métricas como Tempo Médio de Produção de Documentação, Impacto do Processo *DevDocOps* no Tempo de Documentação, Precisão da Documentação, Conformidade entre Documentação e Código em Produção, Tempo de Resposta para Acesso à Documentação e Nível de Interesse na Documentação, descritas na Tabela 4:

Tabela 4 – Perspectiva do Desenvolvedor - métodos de extração de métricas

Método e Objetivo	Métricas Extraídas
Utilização Diária das Diretrizes Propostas: Os desenvolvedores incorporaram cotidianamente as diretrizes sugeridas, refletindo uma integração prática do <i>DevDocOps</i> em seu fluxo de trabalho.	M1 - Eficiência na Produção de Documentação.
	M2 - Impacto do Processo <i>DevDocOps</i> no Tempo de Documentação.
	M3 - Precisão da Documentação.
Feedback Contínuo: O constante feedback proporcionado permitiu ajustes iterativos, promovendo melhorias constantes na produção e qualidade da documentação.	M4 - Conformidade entre Documentação e Código em Produção
	M5 - Tempo de Resposta para Acesso à Documentação.
	M6 - Nível de Interesse na Documentação.

Fonte: Elaborado pela autora.

Para os líderes, a implementação da nova abordagem proporcionou um acompanhamento mais eficiente da entrega das demandas e uma rastreabilidade aprimorada, permitindo a extração de métricas como Impacto do Processo *DevDocOps* no Tempo de Documentação, Nível de Padronização da Documentação, Precisão da Documentação e Nível de Interesse na Documentação, descritas na Tabela 5:

Na análise comparativa, foi possível realizar uma comparação minuciosa entre o funcionamento do processo de documentação anterior e a abordagem *DevDocOps*, identificando benefícios, pontos positivos, desafios e oportunidades de melhorias. Essa abordagem proporcionou uma visão abrangente, possibilitando uma avaliação holística do impacto e eficácia da implementação do *DevDocOps* no contexto específico do projeto, descritas na Tabela 6:

Tabela 5 – Perspectiva do Líder e métodos de extração de métricas

Método e Objetivo	Métricas Extraídas
Acompanhamento Eficiente das Demandas: Os líderes puderam monitorar de forma mais eficaz o progresso das demandas, garantindo maior transparência e controle sobre o ciclo de desenvolvimento.	M1 - Impacto do Processo Dev-DocOps no Tempo de Documentação.
Rastreabilidade Aprimorada: A nova abordagem facilitou uma rastreabilidade mais clara, desde a concepção até a entrega final das demandas.	M2 - Aderência a padrões.
	M3 - Precisão da Documentação.
	M4 - Nível de Interesse na Documentação.

Fonte: Elaborado pela autora.

Tabela 6 – Perspectiva de Análise Comparativa e métodos de extração de métricas

Método e Objetivo	Métricas Extraídas
Comparação com o Processo Atual: Uma análise detalhada comparou o funcionamento do processo de documentação anterior com a implementação do DevDocOps, identificando mudanças, melhorias e desafios.	M1 - Eficiência na Produção de Documentação
	M2 - Aderência a Padrões
	M3 - Identificação de Gargalos e Pontos de Melhoria.
	M4 - Redução de Erros e retrabalhos
Análise de Benefícios e Desafios: A análise comparativa permitiu extrair insights sobre os benefícios alcançados, pontos positivos e negativos da nova abordagem, além de indicar áreas potenciais de aprimoramento.	Satisfação do Desenvolvedor e do Líder

Fonte: Elaborado pela autora.

Com a definição das metas necessárias já realizada, iniciou-se a preparação dos formulários e termos de consentimento destinados aos participantes envolvidos no processo. Esses documentos foram cuidadosamente elaborados para garantir a compreensão clara das métricas a serem avaliadas, assim como para obter o consentimento informado dos participantes que se submeteriam aos procedimentos de teste, e encontram-se disponíveis no Apêndice A deste trabalho.

A elaboração desses formulários e termos de consentimento visou não apenas atender a requisitos éticos e legais, mas também assegurar a transparência e a participação voluntária dos envolvidos no projeto. Esse processo foi conduzido com atenção aos princípios éticos e à proteção dos participantes, estabelecendo uma base

sólida para a condução ética e responsável dos testes propostos. Com os formulários e termos de consentimento devidamente elaborados, procedeu-se à elaboração do plano de testes.

Esta etapa envolveu a determinação da duração do teste, a alocação adequada de recursos humanos, a identificação dos objetos de banco de dados que seriam utilizados e a aplicação de medidas rigorosas para garantir a anonimização eficaz das informações. A parceria estreita com a empresa colaboradora foi fundamental para garantir a viabilidade e o sucesso da execução dos testes, destacando a importância da integração eficaz entre pesquisa acadêmica e prática empresarial.

Tabela 7 – Definição do um plano de testes.

Etapa	Objetivo	Duração	Stakeholders
1	Realizar a apresentação do projeto com os participantes, apresentação da plataforma, liberação de acessos, e como utilizar.	2 horas	Desenvolvedores, Líder Técnico
2	Cadastro das estórias no <i>Board</i> (replicação de pontos e informações), geração de documentos pelos voluntários, responder questionário ao final.	3 horas	Desenvolvedores, Líder Técnico
3	Conferência das documentações disponibilizadas, responder questionário.	2 horas	Líder Técnico

Fonte: Elaborado pela autora.

Foi essencial realizar a apresentação do plano de testes, aguardar a liberação por parte da gestão e, posteriormente, preparar a agenda dos envolvidos. Com a execução da Etapa 1 dos testes, foi promovido o alinhamento da equipe para assegurar a correta utilização da plataforma e do processo proposto. Durante essa fase, foram realizadas atividades como a configuração de acessos, esclarecimento de dúvidas, e outros procedimentos necessários para otimizar a compreensão e adesão ao plano.

As Etapas 2 e 3 foram repetidas ao final de cada ciclo de desenvolvimento, evidenciando a abordagem ágil do processo. Nessas iterações, foram destacados objetos específicos - tabelas e *Calculation Views* que foram desenvolvidos ou modificadas e incorporados ao ambiente produtivo do projeto, estabelecendo um ambiente controlado, no qual os objetos foram submetidos aos testes do processo

proposto, para garantir a qualidade contínua da solução. O ciclo de repetição dessas etapas não apenas permitiu uma avaliação abrangente, mas também possibilitou ajustes incrementais à medida que novos elementos eram introduzidos no ambiente de teste. Essa metodologia contínua de testes não apenas contribuiu para a detecção precoce de possíveis problemas, mas também promoveu uma integração eficaz entre o desenvolvimento e a verificação, reforçando a colaboração entre os indivíduos envolvidos e garantindo a entrega de um produto confiável.

Como resultado dos testes sendo aplicados em três perspectivas distintas - Desenvolvedores, Líder, e Análise Comparativa, obteve-se visões do impacto da utilização por vertentes distintas, contribuindo com a validação da aplicação técnica do projeto.

3.3 Execução

A execução do estudo experimental compreendeu a realização de três etapas distintas, descritas a seguir:

A primeira etapa visava impulsionar a agilidade no processo de documentação, mediante o desenvolvimento de *scripts* capazes de gerar documentação de engenharia de dados, de forma automatizada, a partir de determinados documentos de entrada. Esse processo resultou na obtenção de um documento de saída padronizado, contribuindo para elevar a qualidade do material documentado. Esta etapa foi realizada pela pesquisadora.

A segunda etapa foi delineada com a finalidade de impulsionar o desenvolvimento contínuo de documentação de software e a entrega contínua. Essa etapa envolveu a construção do processo *DevDocOps* dentro de uma ferramenta especializada, capaz de suportar tanto o gerenciamento de projetos ágeis, quanto o fluxo contínuo de integração e entrega (CI/CD). Esta etapa foi realizada pela pesquisadora.

A terceira etapa foi executada com o intuito de realizar a aplicação prática do fluxo proposto em um ambiente de desenvolvimento cuidadosamente controlado. Nesse cenário, foi conduzida uma série de interações na plataforma, capturando

de forma sistemática as respostas dos usuários a cada ciclo de execução do processo. Essa abordagem foi adotada com a finalidade de validar minuciosamente a eficácia e a usabilidade da diretriz proposta, proporcionando *insights* valiosos sobre a experiência prática dos usuários e identificando possíveis melhorias para refinamento contínuo. Essa fase não apenas consolidou a aplicação real do fluxo, mas também contribuiu significativamente para a validação e aprimoramento contínuo da abordagem proposta. Esta etapa foi conduzida pela pesquisadora, e envolveu a participação de três profissionais da empresa parceira: dois desenvolvedores e um líder técnico.

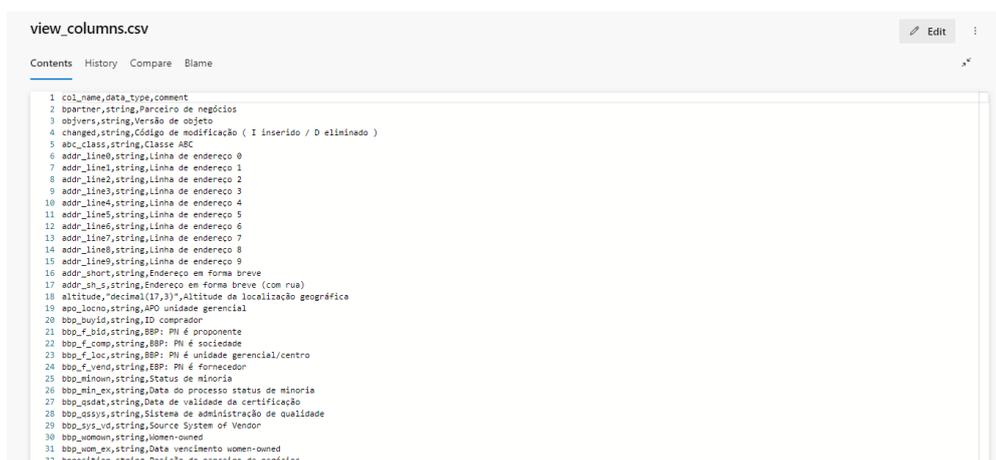
3.3.1 Etapa A - Desenvolvimento de *scripts* de automatização de documentação

Ao definir os objetos de banco de dados específicos que seriam abordados no escopo do projeto, sendo eles estruturas de Tabelas e *Calculation Views*, surgiu a necessidade de desenvolver um *script* especializado. Este *script* foi projetado para realizar uma varredura nos arquivos fontes desses objetos, visando extrair informações relevantes, e, posteriormente, gerar um documento consolidado.

Para realizar a varredura das estruturas de Tabelas com o propósito de gerar dicionários de dados, foi desenvolvido um *script* utilizando a linguagem Python, o script encontra-se disponível no Anexo A. A escolha da linguagem Python proporcionou a agilidade e versatilidade necessárias para manipular os dados, contribuindo significativamente para o processo de geração dos dicionários de dados associados às estruturas de Tabelas identificadas. Este *script* é projetado para receber dois arquivos CSV como entrada, permitindo uma abordagem flexível na extração e análise das informações necessárias.

O primeiro desses arquivos, denominado 'config_doc', disponível no Anexo C, segue um formato fixo e contém informações de configuração a serem incorporadas no documento gerado. Tais informações incluem elementos como título, subtítulo e nome do autor. O segundo arquivo, designado como 'view_columns', é destinado a armazenar a estrutura detalhada da tabela presente no banco de dados, que pode ser obtida por meio da consulta dos metadados da tabela desejada, que

forneem informações sobre os nomes das colunas, tipos de dados, tamanhos e outras propriedades relevantes da tabela.



```
1 col_name,data_type,comment
2 bpartner,string,Parceiro de negócios
3 objvers,string,versão de objeto
4 changed,string,Código de modificação ( I inserido / D eliminado )
5 abc_class,string,Classe ABC
6 addr_line0,string,Linha de endereço 0
7 addr_line1,string,Linha de endereço 1
8 addr_line2,string,Linha de endereço 2
9 addr_line3,string,Linha de endereço 3
10 addr_line4,string,Linha de endereço 4
11 addr_line5,string,Linha de endereço 5
12 addr_line6,string,Linha de endereço 6
13 addr_line7,string,Linha de endereço 7
14 addr_line8,string,Linha de endereço 8
15 addr_line9,string,Linha de endereço 9
16 addr_short,string,Endereço em forma breve
17 addr_sh_s,string,Endereço em forma breve (com rua)
18 altitude,"decimal(17,3)",Altitude de localização geográfica
19 eps_unico,string,EPD unidade gerencial
20 bbp_buyid,string,ID comprador
21 bbp_f_bid,string,EBP: PN é proponente
22 bbp_f_com,string,EBP: PN é sociedade
23 bbp_f_loc,string,EBP: PN é unidade gerencial/centro
24 bbp_f_vend,string,EBP: PN é fornecedor
25 bbp_minow,string,Status de minoria
26 bbp_min_ex,string,Data do processo status de minoria
27 bbp_ssdat,string,Data de validade de certificação
28 bbp_assys,string,Sistema de administração de qualidade
29 bbp_sys_vd,string,Source System of Vendor
30 bbp_womwn,string,women-owned
31 bbp_wom_ex,string,Data vencimento women-owned
32 hnpositlon,string,Posição do parceiro de negócios
```

Figura 5 – Demonstração de Entrada no arquivo de configuração 'view_columns'

Fonte: Elaborado pela autora.

A realização da varredura do arquivo foi construída a partir da biblioteca Pandas³ em Python, que possibilitou a eficiente manipulação e armazenamento dos valores extraídos em um *array*. A escolha dessa biblioteca se deve à sua capacidade de lidar eficazmente com dados tabulares e facilitar a organização das informações recebidas.

A combinação desses dois arquivos possibilitou a personalização eficaz do documento gerado, incorporando tanto as configurações desejadas quanto a estrutura relevante da tabela. Na figura 3.3.1, é possível visualizar a saída obtida ao realizar a execução do *script*.

³ <https://pandas.pydata.org/>

```
# Dicionário de Dados - DIM DATA V3
Dicionário de Dados do desenvolvimento da Tabela VIEW DIM_DATA V3
```

col_name	data_type	comment
bpartner	string	Parceiro de negócios
objvers	string	Versão de objeto
changed	string	Código de modificação (I inserido / D eliminado)
abc_class	string	Classe ABC
addr_line0	string	Linha de endereço 0
addr_line1	string	Linha de endereço 1
addr_line2	string	Linha de endereço 2
addr_line3	string	Linha de endereço 3
addr_line4	string	Linha de endereço 4
addr_line5	string	Linha de endereço 5
addr_line6	string	Linha de endereço 6
addr_line7	string	Linha de endereço 7
addr_line8	string	Linha de endereço 8
addr_line9	string	Linha de endereço 9
addr_short	string	Endereço em forma breve
addr_sh_s	string	Endereço em forma breve (com rua)
altitude	decimal(17,3)	Altitude da localização geográfica
apo_locno	string	APO unidade gerencial
bbp_buyid	string	ID comprador
bbp_f_bid	string	BBP: PN é proponente
bbp_f_comp	string	BBP: PN é sociedade
bbp_f_loc	string	BBP: PN é unidade gerencial/centro
bbp_f_vend	string	BBP: PN é fornecedor
bbp_minown	string	Status de minoria
bbp_min_ex	string	Data do processo status de minoria
bbp_qsdat	string	Data de validade da certificação
bbp_qssys	string	Sistema de administração de qualidade
bbp_sys_vd	string	Source System of Vendor
bbp_womown	string	Women-owned

Figura 6 – Demonstração de Saída do arquivo *Markdown* de Dicionário de Dados

Fonte: Elaborado pela autora.

Para a concepção do *script* capaz de realizar a varredura das *Calculation Views* foi necessário realizar uma análise aprofundada de sua estrutura técnica no SAP HANA. Essa análise envolveu uma compreensão detalhada da organização interna das *Calculation Views*, como as relações entre tabelas, as operações de junção, as agregações, e a lógica subjacente aos cálculos. Durante essa fase de análise, foram identificados os elementos cruciais que compõem uma *Calculation Views*, destacando os pontos críticos para extração de informações relevantes. A compreensão da estrutura XML, que descreve a hierarquia e os relacionamentos, foi essencial para criar um *script*, disponível no Anexo B, capaz de realizar uma varredura precisa e abrangente e, a análise prévia não apenas fundamentou a concepção do *script*, mas também permitiu antecipar desafios potenciais e desenvolver estratégias eficazes para extrair dados técnicos de forma coerente. A sinergia entre a análise da estrutura técnica e o desenvolvimento do *script* foi crucial para garantir a robustez e eficácia do processo de geração de documentação técnica desse objeto.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <calculation:scenario xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:calculation="http://www.sap.com/ndb/6iModelCalculation.ecore"
3 <descriptions defaultDescription="DIM_EXEMPLO"/>
4 <localVariables/>
5 <variableMappings/>
6 <informationModelLayout relativeWidthScenario="48"/>
7 <dataSources>
8 <DataSource id="SCHEMAS.PACKAGE.VIEWS::TABLES_ODS.NOME_DA_VIEW" type="DATA_BASE_TABLE">
9 <viewAttributes allViewAttributes="true"/>
10 <columnObject schemaName="N07_OTHER_STG" columnObjectName="SCHEMAS.PACKAGE.VIEWS::TABLES_ODS.NOME_DA_VIEW"/>
11 </DataSource>
12 <DataSource id="A_NOME_DA_VIEWSSSSSCHEMAS.PACKAGE.VIEWS::TABLES_ODS.NOME_DA_VIEWSS" type="DATA_BASE_TABLE">
13 <viewAttributes allViewAttributes="true"/>
14 <columnObject schemaName="N07_OTHER_STG" columnObjectName="SCHEMAS.PACKAGE.VIEWS::TABLES_ODS.NOME_DA_VIEW"/>
15 </DataSource>
16 <DataSource id="TF_DADOS_NODE_1" type="TABLE_FUNCTION">
17 <resourceUri>LATAM.BR.SALES.SELLOUT.FUNCTIONS::TF_DADOS_NODE_1</resourceUri>
18 </DataSource>
19 <DataSource id="A_N_DSSSSTF_DADOS_NODE_1SS" type="TABLE_FUNCTION">
20 <resourceUri>LATAM.BR.SALES.SELLOUT.FUNCTIONS::TF_DADOS_NODE_1</resourceUri>
21 </DataSource>
22 </dataSources>
23 <calculationViews>
24 <calculationView xsi:type="Calculation:ProjectionView" id="P_CONVERSAO_INDUSTRIA">
25 <descriptions/>
26 <viewAttributes>
27 <viewAttribute id="INDUSTRIA_DATALAKE"/>
28 <viewAttribute id="LAST_ETL_DATE"/>
29 <viewAttribute id="INDUSTRIA_ID"/>
30 </viewAttributes>

```

Figura 7 – Demonstração de Entrada do arquivo XML de *Calculation Views*

Fonte: Elaborado pela autora.

O *script* desenvolvido opera com o arquivo XML associado à *Calculation View*. Utilizando as bibliotecas `lxml`⁴ e `etree`⁵, foi possível mapear de maneira eficaz os nós existentes, identificando suas características, funções e os encadeamentos que permeiam toda a estrutura do objeto. Essa abordagem não apenas simplifica o processo de documentação, mas também assegura uma representação precisa e abrangente das nuances técnicas da *Calculation View*, promovendo uma compreensão detalhada para os envolvidos no processo de análise e manutenção dessas estruturas complexas.

Para a montagem da estrutura do arquivo de saída dos *scripts* e a apresentação das informações de maneira formatada, foi utilizada a biblioteca `Snakemd`⁶. Essa biblioteca é especializada na criação de documentos por meio de *tags*, proporcionando uma abordagem flexível e dinâmica na geração de documentos a partir dos dados processados. O uso conjunto dessas bibliotecas possibilitou uma

⁴ <https://lxml.de/>

⁵ <https://docs.python.org/3/library/xml.etree.elementtree.html>

⁶ <https://www.snakemd.io/en/latest/>

implementação eficaz e organizada do processo de varredura e apresentação de informações, constituindo um arquivo padronizado independente das informações de entradas fornecidas.

```
# Documentação da View DIM_FILIAL_ESTOQUE

0: SCHEMAS.PACKAGE.VIEWS::TABLES_ODS.DATA_BASE_TABLE

1: A_DATA_BASE_TABLE$$$$SCHEMAS.PACKAGE.VIEWS::TABLES_ODS.DATA_BASE_TABLE$$

2: TF_DADOS_NODE_1

3: A_N_D$$$$TF_DADOS_NODE_1$$
## Calculation Node
('0:P_DATA_BASE_TABLE: ', 'Calculation:ProjectionView')
Node Output 0: INDUSTRIA_DATALAKE
 1: LAST_ETL_DATE
 2: INDUSTRIA_ID
 3: LAST_ETL_DATE
 4: INDUSTRIA_DATALAKE
 5: INDUSTRIA_ID
 6: JOIN$LAST_ETL_DATE$LAST_ETL_DATE
 7: LAST_ETL_DATE
 8: GRUPO_FORNECEDOR
 9: COD_FORNECEDOR_int
10: INDUSTRIA_DATALAKE
11: LAST_ETL_DATE
```

Figura 8 – Demonstração de Saída do arquivo *Markdown* de *Calculation Views*

Fonte: Elaborado pela autora.

Com os *scripts* finalizados, foram conduzidos testes de geração de documentação, utilizando objetos do banco de dados em um ambiente local. Essa fase de testes foi crucial para verificar a eficácia e a precisão dos *scripts* desenvolvidos, garantindo que eles pudessem lidar de maneira confiável com diferentes tipos de objetos e fornecer uma documentação consistente, atingindo o objetivo de possibilitar a geração automatizada de documentação de forma padronizada e eficiente.

3.3.2 Etapa B - Definição e construção do fluxo *DevDocOps*

Com o intuito de instituir uma diretriz estruturada para o ciclo de vida do desenvolvimento em Engenharia de Dados, adotando a abordagem *DevDocOps*, foi imprescindível realizar a configuração e parametrização do ambiente *Azure DevOps*. Essa etapa foi fundamental para estabelecer uma infraestrutura robusta

que suporta as práticas de documentação, desenvolvimento e operações de maneira integrada e eficiente. A configuração do ambiente no *Azure DevOps* proporciona as bases necessárias para a implementação de boas práticas ao longo do ciclo de vida do desenvolvimento, contribuindo para a automação, colaboração e rastreabilidade dos processos em Engenharia de Dados.

Para implementar o *Azure DevOps* e garantir seu funcionamento básico, foi essencial realizar a configuração inicial na plataforma, seguindo a realização de 6 passos:

1. **Configuração do Quadro de Acompanhamento:** Criação de um quadro dedicado para o acompanhamento das histórias de desenvolvimento. Este passo proporciona uma visão organizada e visual das tarefas, permitindo uma gestão eficiente do progresso e alinhamento com os objetivos do projeto.
2. **Ativação do Repositório *Azure Repos*:** Habilitar o repositório *Azure Repos*. Essa etapa é fundamental para armazenar, versionar e colaborar no desenvolvimento do código-fonte de maneira segura e integrada.
3. **Configuração do ambiente Wiki:** Habilitar a wiki e vincular ao repositório de armazenamento de código fontes e pipelines. Essa etapa é fundamental para a organização, colaboração eficaz e preservação do conhecimento no ambiente de desenvolvimento
4. **Configuração das Políticas de Código:** Estabelecimento de políticas de código no repositório para promover boas práticas de desenvolvimento.
5. **Configuração de Pipelines de *Build* e *Release*:** Parametrização de pipelines de *build*⁷ e *release*⁸ para automatizar o processo de compilação, execução dos *scripts* de geração de documentação, implantação de código e disponibilização da documentação. Essa automação é crucial para garantir a consistência e a eficiência nas entregas de documentação do projeto.

⁷ O processo de *build* em CI/CD refere-se à compilação e construção do código-fonte de um aplicativo ou sistema.

⁸ O processo de *release* em CI/CD refere-se à distribuição e implantação dos artefatos construídos em ambientes específicos, como servidores de teste, homologação e produção.

6. **Configuração de Notificações e Avisos:** Estabeleceram-se notificações e alertas para manter os membros da equipe informados sobre mudanças importantes, atualizações no código ou eventos relacionados ao ciclo de vida do desenvolvimento.

A implementação da abordagem *DevDocOps* adotou configurações específicas que moldaram as etapas críticas do processo. Os métodos desta proposta foram empregados nas etapas 3, 4 e 5 para personalizar a aplicação da abordagem de acordo com as necessidades particulares desta pesquisa.

Embora os passos de configuração sejam genéricos e aplicáveis a qualquer escopo e utilização da plataforma *Azure DevOps*, nas etapas 3, 4 e 5 foram implementados métodos específicos. Esses métodos compõem as diretrizes propostas por este projeto, proporcionando uma adaptação personalizada da abordagem *DevDocOps* às exigências específicas do projeto em questão. Essa personalização visa otimizar a eficácia do processo e garantir uma integração harmoniosa com as práticas de desenvolvimento e documentação adotadas, aos quais serão descritas nos parágrafos adjacentes.

Na etapa 3, foi realizada a configuração do ambiente *Wiki*, ferramenta interna disponível dentro do *Azure DevOps* como necessidade imperativa no contexto do projeto. Esta ferramenta desempenha um papel na centralização e documentação de informações essenciais, fornecendo um repositório acessível e colaborativo para dados críticos do projeto, práticas, procedimentos e diretrizes. Ao configurar o ambiente *Wiki*, estabelecemos um espaço estruturado para armazenar conhecimentos relevantes, promovendo a transparência, o compartilhamento eficiente de informações e a colaboração contínua entre os membros da equipe. O ambiente não apenas atua como um repositório de documentos estáticos, mas também oferece recursos de edição colaborativa, histórico de versões e capacidades de pesquisa avançadas. Isso possibilita a criação dinâmica e atualização contínua da documentação, refletindo as mudanças e evoluções ao longo do ciclo de vida do projeto. A configuração do ambiente *Wiki* permite a integração natural com outras ferramentas e recursos da plataforma, como as *pipelines*, consolidando a informação em um local centralizado.

Na etapa 4, foram definidas políticas de código no *Azure DevOps* integrando regras que impedem *deploy*⁹ direto na *branch*¹⁰ principal do repositório. Para a execução do *deploy*, foi estabelecido a necessidade de vinculação a uma *branch* secundária, a qual deverá ser associada a uma história ativa no quadro, atribuída a um usuário específico ou sem vinculação a outro usuário. Além disso, também foi estabelecida a condição de requerer a aprovação de um responsável na plataforma antes que o *deploy* seja efetuado na *branch* principal. Essas medidas garantem um controle rigoroso sobre as alterações no código, promovendo uma abordagem segura e validada para a evolução do projeto.

Na etapa 5, foram desenvolvidos pipelines de *build* e *release* personalizados, incorporando os *scripts* que foram desenvolvidos para a criação de documentação automatizada. Essa construção envolveu a utilização de pacotes e a instalação do Python, seguida pela instalação dos pacotes especificados no arquivo 'requirements.txt' por meio do comando `pip install`. A pipeline foi então configurada para realizar a chamada do *script* Python de automação, armazenado no repositório da *branch* principal, possibilitando a execução automatizada das tarefas definidas no código. O código da pipeline foi disponibilizado no Anexo D.

Uma etapa crucial dessa construção foi a captura do arquivo gerado como saída da execução, permitindo o acesso eficiente ao resultado da automação tanto pelo usuário como pela plataforma. Em seguida, foi implementado um passo para copiar esse arquivo para a área de transferência interna no *Azure DevOps* facilitando a manipulação subsequente.

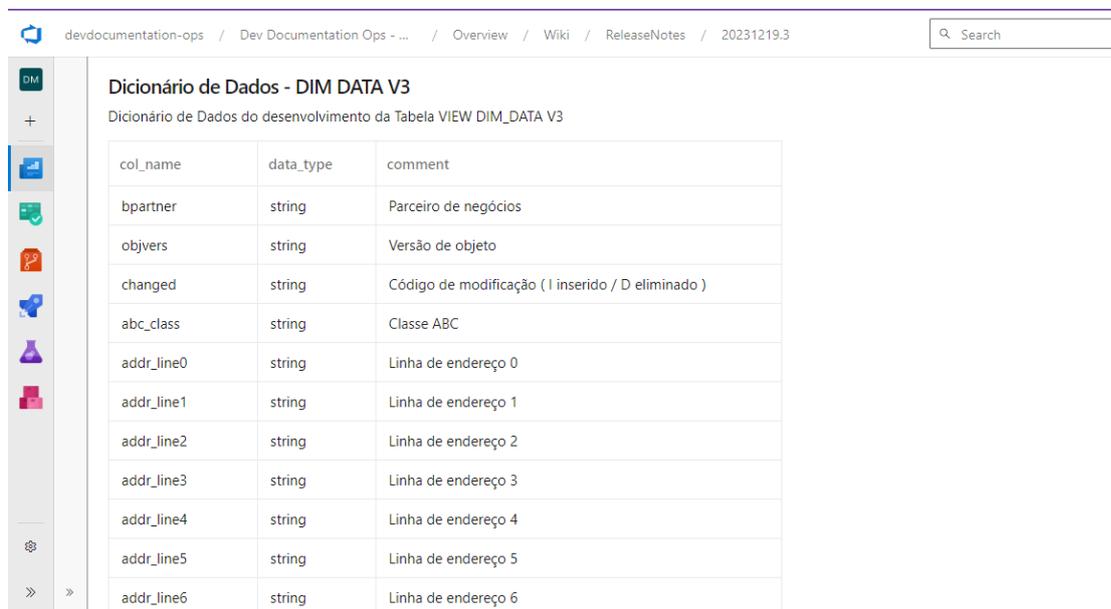
Para finalizar o processo, foi incorporado o componente `WikiUpdaterTask`¹¹, um recurso chave da plataforma, que desempenhou um papel central na integração com a *Wiki*. Utilizando esse componente, foi possível acessar a *Wiki*, e realizar a publicação do arquivo gerado, garantindo a atualização instantânea da documentação com os resultados mais recentes do projeto. Esse fluxo de construção de pipelines

⁹ Refere-se ao processo de levar um software ou aplicativo do ambiente de desenvolvimento para um ambiente operacional, como produção ou outro ambiente de teste.

¹⁰ Refere-se à criação de uma linha de desenvolvimento separada a partir de uma linha principal (*trunk* ou *master*) no sistema de controle de versão.

¹¹ <https://marketplace.visualstudio.com/items?itemName=richardfennellBM.BM-VSTS-WIKIUpdater-Tasks>

não apenas automatiza tarefas cruciais, mas também promove a consistência, rastreabilidade e colaboração eficaz dentro do ambiente de desenvolvimento no *Azure DevOps*.



The screenshot shows a Wiki page titled "Dicionário de Dados - DIM DATA V3". The page content includes a table with the following data:

col_name	data_type	comment
bpartner	string	Parceiro de negócios
objvers	string	Versão de objeto
changed	string	Código de modificação (1 inserido / D eliminado)
abc_class	string	Classe ABC
addr_line0	string	Linha de endereço 0
addr_line1	string	Linha de endereço 1
addr_line2	string	Linha de endereço 2
addr_line3	string	Linha de endereço 3
addr_line4	string	Linha de endereço 4
addr_line5	string	Linha de endereço 5
addr_line6	string	Linha de endereço 6

Figura 9 – Exibição de Documentação Disponibilizada na Wiki

Fonte: Elaborado pela autora.

Os testes de integração do processo foram conduzidos por meio dos acionamentos de execução do pipeline, no qual observou-se a necessidade de otimização para evitar execuções excessivas. Diante desse cenário, foi decidido construir dois pipelines distintos, cada um encarregado da geração específica de cada tipo de documento, definidos no escopo desse projeto, sendo eles Dicionário de dados e Documentação Técnica de *Calculation Views*. Essa abordagem visa aprimorar a eficiência do processo, garantindo que a execução de pipelines seja direcionada apenas quando necessário, resultando em uma gestão mais precisa e econômica dos recursos disponíveis. Com essa implementação, cada pipeline assume a responsabilidade exclusiva pela geração de um tipo particular de documento, contribuindo para a otimização geral do ciclo de vida do desenvolvimento no *Azure DevOps*.

Adicionalmente, ao considerar as limitações identificadas durante a realização dos testes de integração, vislumbra-se uma oportunidade significativa de aprimoramento no processo. Uma proposta para mitigar a necessidade de construir pipelines distintos para tipos específicos de documentos é a inclusão de um varredor de *commit*¹². Esse mecanismo analisaria o tipo de arquivo enviado pelo desenvolvedor e, de forma automática, executaria o pipeline apropriado de acordo com o objeto em questão. Essa funcionalidade não apenas simplificaria a gestão de pipelines, eliminando a necessidade de múltiplos pipelines para diferentes documentos, mas também otimizaria a execução, garantindo que apenas os pipelines relevantes sejam acionados em resposta às alterações efetuadas no repositório. Essa abordagem destaca uma perspectiva promissora para a otimização contínua, alinhada com as práticas de *DevOps* e a eficiência do ciclo de desenvolvimento de engenharia de dados. Vale ressaltar que a promissora natureza dessa iniciativa se destaca, uma vez que, até o momento, a literatura ainda não abraça completamente a concepção de uma abordagem integradora semelhante para esse escopo. No entanto, tal otimização não foi desenvolvida aqui, sendo prevista para trabalhos futuros.

3.3.3 Etapa C - Testes e Condução do Experimento

Com a configuração do ambiente devidamente concluída, a plataforma estava pronta para a execução de testes. Estes foram conduzidos em um ambiente controlado, tendo como base a seleção de alguns objetos desenvolvidos pela equipe, que compuseram o espaço amostral a ser analisado. Ao todo, foram conduzidos testes utilizando sete diferentes objetos de dados desenvolvidos como base para a análise. Esse cenário de testes proporcionou uma avaliação inicial da eficácia das configurações implementadas, garantindo que as funcionalidades fundamentais fossem verificadas em um contexto controlado.

A condução dos testes envolveu uma série de tarefas estratégicas para garantir uma implementação bem-sucedida das diretrizes propostas para Documentação de Engenharia de dados na abordagem *DevDocOps*. Abaixo estão detalhadas as etapas realizadas durante o processo:

¹² Refere-se a uma operação que registra as alterações realizadas no código-fonte em um repositório de controle de versão.

1. **Definição dos Membros da Equipe de Testes:** Definido com a empresa as pessoas da equipe de testes que corresponderiam aos papéis de desenvolvedor júnior, um desenvolvedor sênior e um líder técnico. A diversidade de experiências e habilidades visava abranger diferentes perspectivas durante a avaliação da plataforma.
2. **Apresentação do Projeto para a Equipe Definida:** A equipe de testes foi devidamente apresentada ao projeto, partindo de uma reunião em grupo, com o fim de repassar informações detalhadas sobre os objetivos, metodologia e expectativas relacionadas à implementação da abordagem *DevDocOps* aos voluntários envolvidos.
3. **Liberação de Acessos e Treinamento de Utilização:** Foram providenciados os acessos necessários à plataforma, seguidos por uma sessão de treinamento dedicada. O treinamento abordou os aspectos práticos da utilização da plataforma, garantindo que todos os membros da equipe estivessem familiarizados com os recursos oferecidos.
4. **Cadastro de Tarefas no Azure Boards:** Para organizar o fluxo de trabalho, foram cadastradas as tarefas relacionados aos objetos de desenvolvimento escolhidos para compor a amostra, no *board* da plataforma, delineando as atividades a serem realizadas durante as sessões de testes.
5. **Execução de Sessões da Etapa 4:** Foram realizadas sessões com o grupo de voluntários para a execução de tarefas dentro da diretriz proposto, nos quais os testes de execução e conferência das documentações geradas acompanhassem a evolução do projeto e dos objetos desenvolvidos que compunham a amostra. Ao final de cada execução, os desenvolvedores foram submetidos a um questionário para a realização da coleta de resultados e *feedbacks* do teste.
6. **Entrevista com os Usuários sobre a Experiência de Uso na Plataforma:** A pesquisadora conduziu sessões de bate-papo com a equipe de voluntários para colher impressões mais detalhadas sobre sua experiência na plataforma disponibilizada, explorando áreas de satisfação e eventuais desafios enfrentados.

7. **Conferência das Documentações pelo Líder, Respostas do Questionário:** O líder técnico revisou as documentações geradas na plataforma com base nas entregas realizadas pelos desenvolvedores, e disponibilizadas na *wiki*. Foi realizada a coleta de resultados por meio da aplicação de um questionário, com o objetivo de colher impressões sobre a experiência na plataforma.

8. **Bate Papo com o Líder sobre a Experiência de Uso na Plataforma:** Foi realizada uma última sessão de bate-papo específica com o líder técnico para discutir a experiência na plataforma em comparação com o processo de documentação atual, com o objetivo de encontrar os benefícios percebidos e oportunidades de aprimoramento.

3.4 Resultados

Após conduzir o estudo experimental, focado nas metas e questionamentos delineados na Seção 3.2.3, tornou-se possível analisar de maneira aprofundada as respostas providenciadas pelos desenvolvedores e líderes que participaram ativamente do experimento.

A análise sob cada uma das três perspectivas - Desenvolvedores, Líder e Análise Comparativa - proporcionou resultados específicos que direcionaram a definição de métricas detalhadas, essenciais para validar os resultados do experimento. Este procedimento sistemático permitiu a extração das métricas objetivadas, proporcionando uma avaliação mais precisa do impacto das diretrizes de documentação de engenharia de dados na abordagem *DevDocOps* nas atividades desenvolvidas ao longo do processo.

Com base nas métricas delineadas nas perspectivas do desenvolvedor e líder técnico, conforme apresentado nas Tabelas 4 e 5, estabeleceram-se correlações entre as respostas fornecidas nas questões predefinidas e os indicadores métricos descritos.

A Figura 10 oferece uma visualização detalhada das correlações estabelecidas entre as respostas às questões predefinidas pelos desenvolvedores e os indicadores métricos delineados na perspectiva do desenvolvedor.

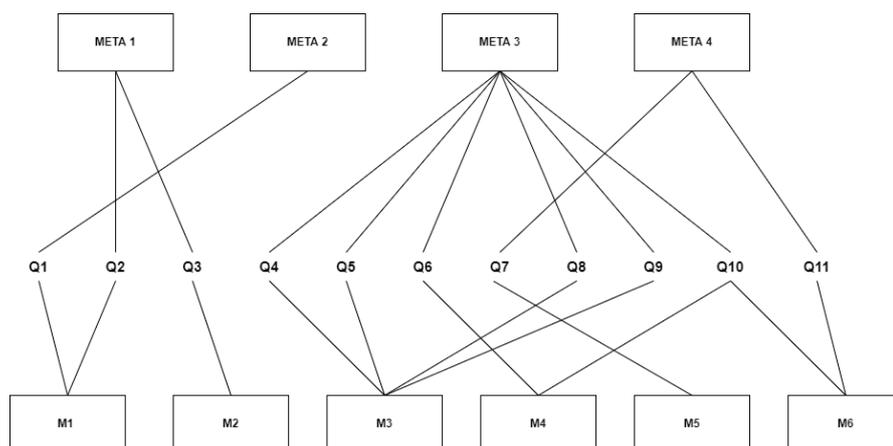


Figura 10 – Relacionamento entre as Metas, Perguntas de Pesquisa e Métricas na perspectiva dos Desenvolvedores.

Fonte: Elaborado pela autora.

Na Figura 11, encontramos uma representação visual das correlações estabelecidas entre as respostas às questões predefinidas pelos líderes técnicos e os indicadores métricos delineados na perspectiva do líder.

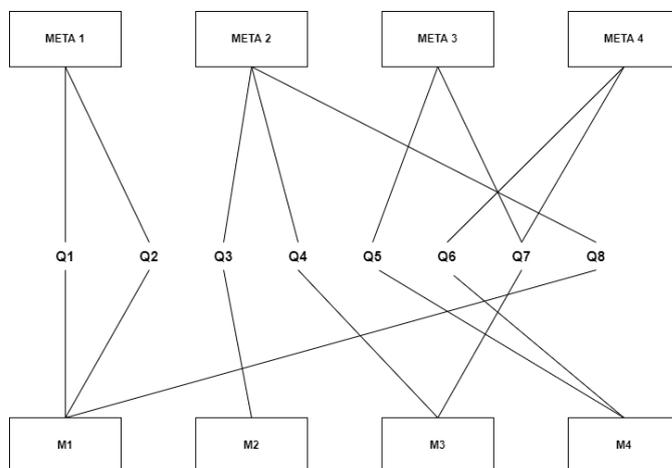


Figura 11 – Relacionamento entre as Metas, Perguntas de Pesquisa e Métricas na perspectiva do Líder

Fonte: Elaborado pela autora.

3.4.1 Análise das Respostas dos Desenvolvedores ao *DevDocOps*: Uma Avaliação Detalhada

Antes de aprofundarmos na análise detalhada das respostas, é crucial compreender o contexto em que essas respostas foram obtidas. O estudo visava avaliar a implementação do *DevDocOps*, uma abordagem que busca otimizar a produção contínua de documentação de engenharia de dados, em um ambiente de desenvolvimento ágil.

Ao longo do processo experimental, dois desenvolvedores, identificados como Desenvolvedor J e Desenvolvedor S, de nível júnior e sênior, respectivamente, participaram ativamente da condução dos testes. Conforme descrito anteriormente, os voluntários foram submetidos a um formulário contendo perguntas específicas destinadas a capturar suas percepções sobre a eficácia da abordagem de *DevDocOps* implementada em termos de produção de documentação, tempo dedicado a tarefas específicas e pareceres individuais dos desenvolvedores sobre o novo processo.

O estudo experimental foi conduzido por meio de sete execuções distintas, cada uma baseada em diferentes objetos desenvolvidos. Essa abordagem diversificada permitiu uma análise abrangente e uma compreensão mais holística do impacto causado nas atividades diárias de desenvolvimento e na produção de documentação. As execuções envolveram situações variadas, proporcionando uma visão mais completa das possíveis nuances e efetividade do novo processo em diferentes contextos de desenvolvimento de engenharia de dados.

A partir da exploração dos resultados fornecidos, serão detalhadas a análise aprofundada sobre o impacto do *DevDocOps* nas atividades diárias de desenvolvimento e na produção de documentação.

- **Pergunta 1:** Foi produzida ou atualizada documentação nessa estória?

As respostas obtidas para esta pergunta referem-se à Questão 1 (Q1) do GQM, contudo, todas as estórias testadas em ambiente controlado envolveram o desenvolvimento de objetos e a correspondente produção ou atualização de documentação, portanto, o resultado obtido não influenciou significativamente nas métricas desejadas, sugerindo que a produção de documentação estava

intrinsecamente relacionada às atividades de desenvolvimento nas estórias testadas.

- **Pergunta 2:** Quantidade de Tempo Gasto (horas) com a Documentação (tempo para geração automática, revisões)

As respostas obtidas para esta pergunta referem-se à Questão 2 (Q2) do GQM, que aborda o tempo dedicado à documentação. Os dados descritos na Tabela 8 estão relacionados à análise do tempo dedicado à documentação, especificando o tipo de documentação e o desenvolvedor envolvido.

Tabela 8 – Análise do Tempo Gasto na Documentação por Tipo e Desenvolvedor (GQM: Q1)

Tarefa	Desenvolvedor	Tempo gasto	Objeto
1	J	10 minutos	Dicionário de dados
2	S	5 minutos	Dicionário de dados
3	J	< 10 minutos	Dicionário de dados
4	S	5 minutos	Dicionário de dados
5	J	15 a 25 minutos	<i>Calculation View</i>
6	S	20 minutos	<i>Calculation View</i>
7	J	10 minutos	<i>Calculation View</i>

Fonte: Elaborado pela autora.

A análise dos resultados permitiu extrair informações sobre como o tempo é distribuído entre diferentes tipos de documentação, permitindo uma compreensão mais profunda do processo e identificando áreas de possível melhoria ou eficiência, descritas abaixo:

- Os resultados obtidos indicam uma eficiência notável na documentação do tipo Dicionário de Dados, especialmente para o Desenvolvedor S.
- A variação de tempo na documentação do tipo *Calculation View* sugere que essa categoria pode apresentar desafios variáveis de acordo com o tamanho e complexidade do objeto desenvolvido.
- A consistência nas execuções do Desenvolvedor J pode indicar uma abordagem sistemática e estável.

- **Pergunta 3:** Se não tivesse sido utilizado o processo *DevDocOps*, você estimaria ter gastado quanto tempo?

As respostas obtidas para esta pergunta referem-se a Questão 3 (Q3) do GQM, que aborda o tempo dedicado à documentação. Os dados descritos na Tabela 9 estão relacionados à estimativa de tempo que seria gasto, dedicado à documentação, caso o processo fosse realizado da forma tradicional, especificando o tipo de documentação e o desenvolvedor envolvido.

Tabela 9 – Estimativa de Tempo Gasto na Documentação por Tipo e Desenvolvedor (GQM: Q3)

Tarefa	Desenvolvedor	Tempo estimado sem DevDocOps	Objeto
1	A	45 mins a 1 hora	Dicionário de dados
2	B	45 mins a 1 hora	Dicionário de dados
3	A	2 horas	Dicionário de dados
4	B	2 horas	Dicionário de dados
5	A	2 a 3 horas	<i>Calculation view</i>
6	B	2 horas	<i>Calculation view</i>
7	A	2 a 3 horas	<i>Calculation view</i>

Fonte: Elaborado pela autora.

Os resultados obtidos sugerem que o *DevDocOps* pode desempenhar um papel crucial na redução do tempo estimado para a documentação, mostrando uma consistência nas percepções dos desenvolvedores entre si, indicando uma eficiência notável na implementação do processo. Esses resultados foram traduzidos em:

- a. **Consistência nas Estimativas:** Tanto o Desenvolvedor J quanto o Desenvolvedor S possuem visões de estimativas relativamente semelhantes entre si para o mesmo tipo de documentação. Ambos os desenvolvedores, em suas diferentes execuções, mostraram concordância na estimativa de tempo.
- b. **Redução de tempo significativa:** Comparando as estimativas da pergunta 3 com os tempos reais gastos (da pergunta 2), é evidente que o uso do processo *DevDocOps* resultou em uma redução considerável no tempo dedicado à documentação. As estimativas sem o *DevDocOps* geralmente

eram mais altas do que os tempos reais registrados, indicando uma eficiência clara, principalmente na produção da documentação, com a implementação das novas diretrizes.

- c. **Variação nas Estimativas:** A existência na variação das estimativas deve-se ao fato de cada objeto desenvolvido possuir um nível de complexidade diferente, afetando diretamente na quantidade de tempo necessário para produzir documentação manualmente.
- d. **Impacto Diferenciado por Tipo de Documentação:** A estimativa de tempo varia conforme o tipo de documentação. Para o Dicionário de Dados, as estimativas tendem a ser mais homogêneas - na forma manual, para a construção de um dicionário de dados, as informações e metadados da tabela são consultados no sistema, e os resultados obtidos são transcritos para um arquivo Excel, seguindo por tarefas de padronização do documento (aplicação de fontes, cores, bordas, formatação de texto, entre outros). Para a *Calculation View*, há uma gama mais ampla de estimativas, pois sua documentação varia de acordo com o seu tamanho e complexidade - na forma manual, a documentação de *Calculation Views* serve como Documento Funcional do objeto, sendo escrita no Word, seu conteúdo é preenchido com imagens de cada nó existente, seguindo o fluxo dos dados até a saída final.

- **Pergunta 4:** O Objeto em desenvolvimento já possui documentação?

As respostas obtidas para a pergunta 4 (Q4) do GQM estão relacionadas à presença de documentação nos objetos em desenvolvimento, com o objetivo de compreender o impacto do versionamento de documentação de dados. Os dados descritos na Tabela 10 estão associados à avaliação sobre se o objeto desenvolvido já possui documentação, especificando o tipo de documentação que necessita ser entregue e identificando o desenvolvedor envolvido.

Com base na análise dos resultados brutos obtidos, foi possível compreender quais os impactos causados pela influência das diretrizes na prática de documentação no ambiente de desenvolvimento ágil, principalmente no que tange a avaliação da existência e concepção de documentação para objetos de

Tabela 10 – Resultados da Avaliação da Presença de Documentação nos Objetos em Desenvolvimento (GQM Q4)

Iteração	Resultado	Desenvolvedor	Objeto
1	Não	Desenvolvedor J	Dicionário de Dados
1	Não	Desenvolvedor S	Dicionário de Dados
2	Sim	Desenvolvedor J	Dicionário de Dados
2	Sim	Desenvolvedor S	Dicionário de Dados
3	Sim	Desenvolvedor J	<i>Calculation View</i>
3	Não	Desenvolvedor S	<i>Calculation View</i>
4	Sim	Desenvolvedor J	<i>Calculation View</i>

Fonte: Elaborado pela autora.

dados que sofrem alterações. Os resultados necessários para a compreensão do impacto do processo na produção contínua de documentação, sendo descritos em:

- a. **Variação na Presença de Documentação:** Os desenvolvedores apresentaram variação na presença de documentação nos objetos em desenvolvimento. Isso sugere que a abordagem para documentação pode variar entre diferentes execuções e tipos de objetos, influenciando a decisão de documentar ou não.
 - b. **Fortalecimento da Abordagem *DevDocOps*:** Os resultados indicam que o uso do *DevDocOps* pode estar fortalecendo a presença de documentação nos objetos em desenvolvimento. Desenvolvimentos que adotaram o *DevDocOps* tendem a ter mais objetos documentados, enquanto aqueles que não adotaram têm variação nas respostas.
 - c. **Necessidade de Reforço na Documentação:** A resposta “Não” para a presença de documentação em alguns objetos indica a necessidade da equipe de compreender o nível de carência de documentação dos objetos em desenvolvimento e produtivos.
- **Pergunta 5:** Caso sua resposta tenha sido “sim” para a questão 4, responda: Você precisou consultar a documentação anterior deste objeto para realizar a tarefa?

As respostas coletadas estão associadas à Questão 5 (Q5) do GQM, a qual explora a necessidade de acesso a documentação anterior em casos em que a resposta anterior foi afirmativa. Os dados apresentados na Tabela 11 relacionam-se a necessidade de utilizar documentações existentes de versões anteriores do objeto, descrevendo apenas as iterações que possuam documentação criada, conforme informado na resposta fornecida à Q4, discriminando o tipo de documentação e identificando os desenvolvedores envolvidos.

Tabela 11 – Resultados da Necessidade de Consulta à Documentação Anterior (GQM Q5)

Iteração	Resultado	Desenvolvedor	Objeto
2	Sim	Desenvolvedor J	Dicionário de Dados
2	Não	Desenvolvedor S	Dicionário de Dados
3	Não	Desenvolvedor J	<i>Calculation View</i>
4	Não	Desenvolvedor J	<i>Calculation View</i>

Fonte: Elaborado pela autora.

Essa análise proporcionou maiores entendimentos sobre a efetividade da documentação anterior na execução das tarefas, destacando situações em que a referência à documentação prévia foi essencial e em que contextos ela não se mostrou necessária, sendo descritos em:

- a. Padrões de Consulta: Observou-se que, quando os desenvolvedores precisaram consultar a documentação anterior (Questão 4), a necessidade variou entre os tipos de objetos desenvolvidos. Para os dicionários de dados, em algumas vezes foi necessário consultar a documentação anterior para entendimento da tarefa e mudanças a serem desempenhadas na demanda de desenvolvimento atual. Quanto às *Calculation Views*, devido a documentação anterior se tratar de documentação funcional, exibindo o mesmo fluxo de dados gráfico visível para o desenvolvedor no momento da edição, não foi necessário nenhum tipo de consulta.
- b. Efetividade da Documentação Atual: Nos casos em que a resposta à Questão 4 foi “não”, destaca-se a possível efetividade da documentação atual, sugerindo que os desenvolvedores conseguiram realizar as tarefas sem recorrer à documentação anterior. Essa observação pode indicar

um bom alinhamento entre a documentação existente e as necessidades atuais de desenvolvimento.

- c. Impacto na Produtividade: A análise dos resultados permite avaliar o impacto da necessidade de consulta à documentação anterior na produtividade dos desenvolvedores. Situações em que a consulta é necessária podem influenciar o tempo dedicado à execução de tarefas devido ao tempo gasto para localizar o documento, e interpretá-lo.
- d. Identificação de Melhorias: A identificação de casos em que a consulta à documentação anterior não foi necessária pode indicar áreas da documentação que podem ser otimizadas ou simplificadas. Isso sugere oportunidades de melhoria na elaboração da documentação, visando torná-la mais acessível e eficaz para os desenvolvedores.

Os resultados obtidos auxiliaram na compreensão sobre a relação entre a necessidade de consultar a documentação anterior e os diferentes contextos de desenvolvimento, fornecendo subsídios para o planejamento de melhorias futuras no processo de documentação.

- **Pergunta 6:** Caso sua resposta tenha sido “sim” para a questão 4, responda: A documentação anterior estava atualizada com a mesma versão do objeto em produção?

As respostas apresentadas se relacionam com a Q6 do GQM, abordando a sincronização entre a documentação anterior e a versão do objeto em produção. Os resultados descritos na Tabela 12 proporciona apontamentos cruciais sobre a relevância da atualização da documentação em relação à evolução dos objetos de dados, descrevendo apenas as iterações que possuam documentação criada, conforme informado na resposta fornecida à Q4.

Os resultados obtidos foram essenciais para a compreensão sobre a relação entre a documentação anterior e as versões em produção, destacando casos em que a atualização é aplicada, não se aplica ou está ausente. A análise dos dados obtidos oferecem um panorama abrangente da eficácia do processo de manutenção da documentação em paridade com a evolução dos objetos de dados, sendo descritas em:

Tabela 12 – Sincronização entre Documentação Anterior e Versão em Produção (GQM Q6)

Iteração	Resultados	Desenvolvedor	Objeto
2	Sim	Desenvolvedor J	Dicionário de Dados
2	Não	Desenvolvedor S	Dicionário de Dados
3	Sim	Desenvolvedor J	<i>Calculation View</i>
4	Sim	Desenvolvedor J	<i>Calculation View</i>

Fonte: Elaborado pela autora.

- a. Casos em que a Atualização se Aplica: os casos em que a resposta foi “Sim”, observa-se uma efetiva manutenção da documentação atualizada com a versão em produção. Isso pode indicar práticas de documentação eficientes e/ou a preocupação em manter a coesão entre o código em execução e a documentação correspondente.
- b. Casos em que a Atualização Não se Aplica: No caso sugerem que, podem haver circunstâncias nas quais a atualização da documentação pode não ser uma prioridade ou que não houve possibilidade de documentar.
- c. Impacto na Confiabilidade da Documentação: A análise desses resultados fornece a percepção sobre a existência de confiabilidade da documentação, considerando a sincronização com as versões em produção. Com a necessidade de sincronização da documentação, podem indicar um esforço contínuo para manter a documentação alinhada com as mudanças nos objetos de dados.
- d. Identificação de Gargalos ou Desafios: A presença de casos em que a atualização não se aplica pode indicar áreas em que os processos de documentação podem ser ajustados para melhor atender às necessidades específicas de diferentes tipos de objetos. Isso aponta para a possível identificação de gargalos ou desafios na gestão da documentação.

Os resultados descritos oferecem um panorama abrangente da eficácia do processo de manutenção da documentação em paridade com a evolução dos objetos de dados desenvolvidos.

- **Pergunta 7:** Caso sua resposta tenha sido “sim” para a questão 4, responda: Foi fácil acessar ou localizar a documentação?

Os resultados coletados para a pergunta são relacionados à Q7 do GQM, discorrendo sobre a facilidade de acesso ou localização da documentação, proporcionando percepções sobre a usabilidade e a eficiência do processo de recuperação de informações exibidos na Tabela 13:

Tabela 13 – Facilidade de Acesso à Documentação (GQM Q7)

Iteração	Resultado	Desenvolvedor	Objeto
2	Sim	Desenvolvedor J	Dicionário de Dados
2	Sim	Desenvolvedor S	Dicionário de Dados
3	Não	Desenvolvedor J	<i>Calculation View</i>
4	Sim	Desenvolvedor J	<i>Calculation View</i>

Fonte: Elaborado pela autora.

Os dados destacam a percepção dos desenvolvedores sobre a facilidade de acessar ou localizar a documentação, viabilizando o direcionamento de ações para otimizar a experiência do desenvolvedor em relação à acessibilidade da documentação, que foram traduzidos nas seguintes análises:

- a. **Facilidade de Acesso para a Maioria dos Casos:** Os desenvolvedores indicaram que foi fácil acessar ou localizar a documentação existente. Isso sugere que, em geral, o processo adotado para armazenamento e recuperação de informações foi eficiente.
- b. **Desafios Identificados em Casos Específicos:** Em um dos casos o desenvolvedor indicou que não foi fácil acessar ou localizar a documentação. Isso pode indicar possíveis desafios na estrutura ou organização da documentação para esse tipo específico de objeto.
- c. **Avaliação da Experiência do Desenvolvedor:** A facilidade de acesso à documentação desempenha um papel crucial na eficiência do desenvolvimento. Os resultados fornecem uma avaliação direta da experiência do Desenvolvedor J buscar informações necessárias para realizar tarefas específicas.

Os resultados indicam que, de maneira geral, os dados são facilmente acessíveis pela equipe de desenvolvimento. Contudo, identificaram-se obstáculos em situações específicas. Nota-se que as atividades realizadas pelo Desenvolvedor S, devido ao seu papel de nível sênior, têm um impacto mais significativo na equipe. Assim, o tempo despendido na localização de arquivos pode ter implicações prejudiciais nas entregas a longo prazo. Essa observação destaca a importância de abordagens específicas para garantir a eficiência, especialmente em casos que envolvem membros mais experientes da equipe.

- **Pergunta 8:** Foi necessário desenvolver alguma documentação de forma manual? Se sim, descreva.

As respostas obtidas para a pergunta sobre a necessidade de desenvolver documentação manualmente referem-se à Questão 8 (Q8) do GQM, que aborda a eficiência do processo *DevDocOps* na automação da geração de documentação.

A maioria das execuções não demandou a criação manual de documentação, indicando que as diretrizes atenderam satisfatoriamente à automação desse aspecto. Entretanto, na execução do Desenvolvedor J, na iteração 4, foi necessário desenvolver manualmente uma documentação funcional, considerando que a gerada pelo processo é predominantemente técnica.

- **Pergunta 9:** A documentação precisou de ajustes após ter sido gerada? Se sim, quais?

As respostas obtidas para a pergunta referem-se à Questão 9 (Q9) do GQM, que busca avaliar a qualidade inicial da documentação produzida pela automatização.

A maioria das execuções não demandou ajustes na documentação gerada, indicando que o documento entregue na automatização atendeu adequadamente aos requisitos de formato e conteúdo.

Na execução do Desenvolvedor J na iteração 2, houve uma exceção no qual foram necessários ajustes nas descrições e formatação do documento, visando melhorar a visualização e hierarquia dos nós na documentação da *Calculation View*. Essas adaptações visavam tornar o conteúdo mais compreensível e

facilitar a interpretação dos pontos importantes e demonstraram-se necessárias de acordo com o nível de complexidade do objeto e suas características específicas.

- **Pergunta 10:** Quão fácil foi a utilização da plataforma?

A avaliação da facilidade de utilização da plataforma constitui a Questão 10 (Q10) do GQM, visando compreender a experiência dos desenvolvedores durante a execução do processo *DevDocOps*. A análise considera uma escala de 1 a 5, onde 1 indica dificuldade extrema e 5 representa facilidade máxima. Os resultados demonstram uma alta satisfação geral, com a maioria das respostas atribuindo a pontuação máxima.

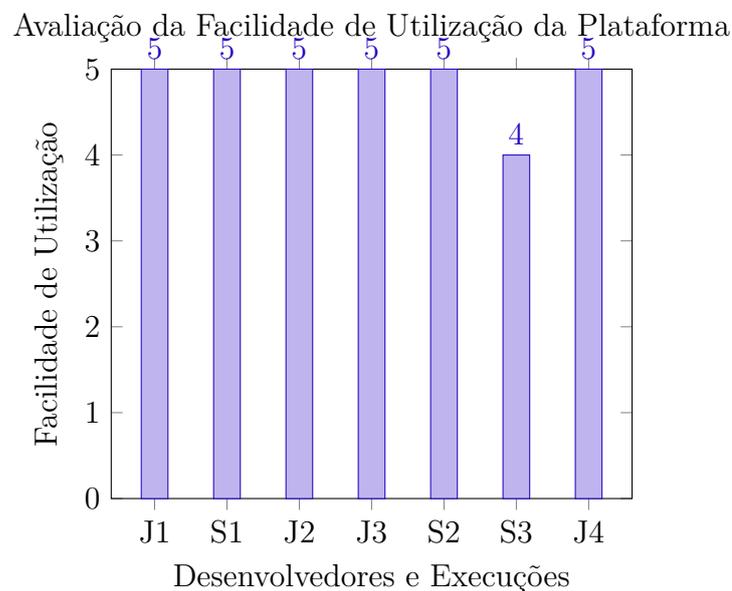


Figura 12 – Avaliação da Facilidade de Utilização da Plataforma

Fonte: Elaborado pela autora.

Os desenvolvedores atribuíram principalmente a nota 5, indicando que a plataforma foi percebida como intuitiva e fácil de aplicar. A única resposta com a pontuação 4 foi justificada pela falta de recordação dos passos entre os testes, o que revela que, mesmo após um intervalo de tempo, a plataforma se mostrou acessível e amigável. Esse resultado destaca a robustez da interface,

sugerindo que, uma vez familiarizados, os usuários consideraram a plataforma fácil de usar.

- **Pergunta 11:** Houve algum problema na utilização do fluxo *DevDocOps*?

A avaliação da facilidade de utilização da plataforma constitui a Questão 9 (Q9) do GQM, que aborda possíveis problemas na utilização do fluxo *DevDocOps*. Todas as respostas obtidas foram “Não”, demonstrando uma experiência geral positiva e sem contratempos significativos durante a aplicação do fluxo. Os participantes relataram uma execução suave do processo, indicando uma implementação bem-sucedida e sem obstáculos notáveis.

- **Pergunta 12:** Com o novo processo, a documentação tornou-se obrigatória. Como você avalia os benefícios e problemas dessa entrega contínua de documentação?

Ao explorar as percepções dos desenvolvedores sobre a implementação do *DevDocOps* e a tornar a documentação uma prática obrigatória, foi descoberto como essa mudança afetou suas experiências e produtividade. A Pergunta 12 buscou entender como os envolvidos percebem os benefícios e possíveis desafios associados à entrega contínua de documentação.

A seguir, serão apresentadas as impressões dos desenvolvedores A e B, fornecendo uma visão abrangente sobre como a introdução dessa prática impactou positivamente suas atividades diárias e, ao mesmo tempo, considerou os eventuais desafios percebidos.

Impressões do Desenvolvedor J:

- a. “Não vejo problemas, e como benefícios, destaco principalmente a economia de tempo, considerando que o processo se tornou mais simples e constante, centralizando e padronizando automaticamente as documentações.”
- b. “O processo se torna mais padronizado e ágil, proporcionando grande auxílio em projetos ágeis.”

- c. “A documentação se torna mais facilmente padronizável, permitindo que o tempo que seria gasto para documentar seja direcionado para outras atividades.”

Impressões do Desenvolvedor S:

- a. “Acredito que os benefícios são a praticidade e agilidade nas atualizações das documentações; não vejo grandes problemas, apenas um trabalho repetitivo e constante.”
- b. “Impacta positivamente no processo de documentação, gerando um alto ganho de tempo e performance.”

Essas impressões indicam uma percepção positiva em relação aos benefícios do novo processo, destacando ganhos de eficiência, padronização e agilidade na atualização da documentação, sendo cruciais para o estabelecimento das avaliações do impacto do *DevDocOps* no contexto da equipe de desenvolvimento.

- **Pergunta 13** - Quais os benefícios e dificuldades que você constatou sobre a forma de entrega (disponibilização, acesso e visualização) da documentação na plataforma?

A última pergunta realizada no questionário visa aprofundar a compreensão sobre como a forma de entrega da documentação na plataforma foi percebida pelos desenvolvedores, destacando tanto os benefícios identificados quanto eventuais desafios enfrentados. A seguir, serão apresentadas as percepções dos desenvolvedores J e S sobre esses elementos, oferecendo uma visão abrangente sobre a usabilidade da plataforma no contexto da entrega contínua de documentação no âmbito do *DevDocOps*.

Impressões do Desenvolvedor J:

- a. “O único ponto é o costume a plataforma por falta de conhecimento porém é simples e intuitivo da mesma forma.”
- c. “A facilidade da junção dos processos no *DevOps* facilita a rastreabilidade do objeto de forma que eu consiga chegar até o documento esperado pela tarefa ou pela wiki.”

- d. “Não encontrei nenhuma dificuldade exceto em relação ao documento em si, que sem as formatações estava um pouco complexo de se entender e um pouco confuso as hierarquias de título e de objetos. Quanto a localização não vejo nenhum ponto de atenção, são bem fáceis de se utilizar e intuitivos.”

Impressões do Desenvolvedor S:

- a. “Facilidade e praticidade para atualização da Documentação.”
- b. “Facilidade e praticidade para a disponibilização, acesso e visualização das documentação pré e pós alterações.”
- c. “Acredito que é relativamente fácil realizar as atividades, porém pode gerar dificuldades para quem nunca trabalhou com a plataforma.”

Ao analisar as respostas dos participantes, é possível vislumbrar aspectos cruciais relacionados à disponibilização, acesso e visualização da documentação, sendo descritos abaixo:

- Adaptação à Plataforma: O Desenvolvedor J destaca o desafio inicial relacionado ao hábito de utilização da plataforma, atribuído à falta de familiaridade. Contudo, reconhece que a adaptação ocorre de forma gradual e que a plataforma é simples e intuitiva. A familiaridade crescente com a plataforma é um padrão identificado entre os desenvolvedores, indicando que, com o tempo, os desafios iniciais de adaptação tendem a ser superados.
- Facilidade de Utilização e Acesso: O Desenvolvedor J menciona a facilidade de se acostumar com a plataforma, expressando confiança e familiaridade crescentes ao longo do tempo. A junção dos processos no DevOps é destacada como facilitadora da rastreabilidade do objeto até o documento desejado. O mesmo também é reconhecido pelo Desenvolvedor S, que destacou a facilidade e praticidade para a atualização da documentação.
- Considerações sobre a Integração de Recursos: A integração dos processos no *DevDocOps* é reconhecida como um ponto positivo, fornecendo uma

visão clara e rastreável da relação entre os objetos e seus documentos correspondentes.

- Considerações sobre Formatação: O Desenvolvedor J observa uma dificuldade pontual relacionada à formatação dos documentos, indicando que, sem as formatações, alguns documentos tornam-se complexos de entender, especialmente em relação à hierarquia de títulos e objetos. A formatação dos documentos emerge como um aspecto sensível, sugerindo a importância de garantir clareza e consistência na apresentação, por meio de configurações de parametrização do *template*, especialmente para facilitar a compreensão de hierarquias e objetos.
- Dificuldades para Iniciantes: O participante menciona que, embora a plataforma seja relativamente fácil para realizar as atividades, pode gerar dificuldades para aqueles que não têm experiência prévia com a plataforma. A consideração das respostas destaca a necessidade de fornecer suporte e treinamento adequados para garantir uma transição suave e eficiente para aqueles que estão iniciando o uso da plataforma.

3.4.2 Análise das Respostas do Líder ao *DevDocOps*: Uma Avaliação Detalhada

Com base na efetiva implementação das diretrizes de documentação de engenharia de dados na abordagem *DevDocOps* no ambiente controlado, cabe ressaltar que o líder desempenhou um papel fundamental na validação das tarefas executadas pelos desenvolvedores. Sua responsabilidade incluiu não apenas a verificação da qualidade e precisão das atividades realizadas, mas também a supervisão e autorização para as respectivas liberações. Nesse contexto, o líder atuou como um ponto crucial de controle, garantindo que as práticas e diretrizes do *DevDocOps* fossem aplicadas de maneira consistente e eficaz. Esta seção oferecerá uma análise minuciosa das respostas do líder, destacando suas percepções e contribuições para o processo de implementação.

O questionário ao qual o líder foi submetido consistiu em 8 perguntas, cada uma correlacionada, em ordem sequencial, com as questões previamente definidas

no GQM. As respostas desse formulário foram coletadas ao final de todo o processo de execução dos desenvolvedores, visando explorar a visão abrangente do líder sobre todos os aspectos e vertentes do processo.

A análise aprofundada das respostas do líder proporcionou uma compreensão mais detalhada sobre a percepção e a experiência dele em relação à implementação do *DevDocOps*. Os resultados incluem avaliações sobre a eficácia do processo na coordenação e validação das tarefas realizadas pelos desenvolvedores, bem como possíveis desafios e benefícios identificados do ponto de vista do líder. A avaliação detalhada, descrita abaixo, contribuirá para uma visão mais completa do impacto do *DevDocOps* nas atividades da equipe, fornecendo informações valiosas para aprimoramentos futuros e otimizações no processo.

- **Pergunta 1:** A agilidade da documentação foi impactada positiva ou negativamente com a implantação desse processo?

“A documentação concebida pelo o projeto proposto demonstra qualidade e confiabilidade das informações obtidas dos modelos de dados, além dar visibilidade a determinado níveis de detalhamento técnicos de forma rápida e ágil assim melhorando o processo de documentação da equipe de Projetos. Adicionalmente além de facilitar a equipe a passagem de conhecimento onde agiliza o onboarding para novos membros e assim promovendo colaborações dentro da equipe, além de reduzir erros decorrentes da falta de clareza.”

A análise da resposta fornecida para a Questão 1 (Q1) revela um impacto positivo na agilidade da documentação com a implementação do *DevDocOps*. A documentação passou a ser percebida como ágil, proporcionando qualidade, visibilidade e facilitando a passagem de conhecimento na equipe.

- **Pergunta 2:** O que melhorou e o que piorou em relação à agilidade do desenvolvimento das estórias?

“A geração do artefato é rápida, em nível detalhamento que passa despercebido em determinados momentos considerando o modelo tradicional. além de permitir maior agilidade a equipe técnica a focar em outros entregáveis ao longo do projeto. Único item que é necessário monitoramento e o aumento

da governança de revisões dos artefatos, pois ao longo projeto possa surgir cenários não mapeados até então.”

A análise da resposta fornecida para a Questão 2 (Q2) do GQM revela a agilidade no desenvolvimento de histórias foi um dos pontos abordados pelo líder. Observa-se uma melhoria significativa na geração rápida de artefatos, permitindo maior foco em outras atividades.

Pode-se destacar como ponto de atenção, a necessidade de monitoramento das entregas e aumento da governança nas revisões dos artefatos.

- **Pergunta 3:** A quantidade de documentação entregue na *sprint* aumentou ?

“Sim, Aumento das documentações é visível a cada entregavel rapidamente é possui os artefatos técnicos, além de termos a documentação clara e consequentemente o foco da equipe DEV está no desenvolvimento e consequentemente mas entregavel dentro da capacidade.”

A análise da resposta fornecida para a Questão 3 (Q3) do GQM revela que a quantidade de documentação entregue durante a *sprint* foi explorada para compreender o impacto da implementação, identificando o aumento notável na quantidade de documentação, proporcionando clareza e foco efetivo na capacidade da equipe.

- **Pergunta 4:** Em termos de padronização da documentação, houve melhoria?

“Sim, é percebido que os artefatos segue padronizações em formatações em geral, além de reduzir o tempo da equipe na sua confecção e reduz erros decorrentes da falta de clareza. Além de demonstrar o documento objetivo e com as informações bem apresentadas visualmente no qual gera bastante confiança para equipe e consequente o cliente.”

A análise da resposta fornecida para a Questão 4 (Q4) do GQM revela como a padronização da documentação foi avaliada quanto à formatação e clareza. A percepção do líder evidenciou uma melhoria na padronização, reduzindo o tempo de confecção e minimizando erros.

- **Pergunta 5:** Em comparação ao processo de documentação anterior com o novo processo, a documentação entregue cumpre a necessidade técnica?

“Sim, em comparação a métodos tradicionais é possível termos ganhos significativos de tempo, padronização e detalhamento técnico dos artefatos e modelos de dados. Além dar visibilidade a equipe e liderança técnica que estão sendo cumpridas a documentação de cada artefato técnico desenvolvido. Para liderança técnica ajuda na mudança de cultura onde é possível mostrar a importância da documentação não apenas como uma tarefa, mas como um investimento estratégico, no qual incentivam e apoiam a equipe a priorizar a criação de documentação em todas as fases do desenvolvimento.”

A análise da resposta fornecida para a Questão 5 (Q5) do GQM revela a percepção sobre como a documentação atende às necessidades técnicas foi analisada. A percepção do líder trouxe uma comparação do método utilizado aos métodos tradicionais, observam-se ganhos significativos em tempo, padronização e detalhamento técnico.

- **Pergunta 6:** Com o novo processo, a documentação tornou-se obrigatória. Como você avalia os benefícios e problemas dessa entrega contínua de documentação?

“Como liderança vejo a documentação contínua sendo ‘viva’ no ciclo do projeto, onde a documentação deve ser tratada como parte do ciclo contínuo, caso haja novas mudanças e inclusões, a documentação sensibiliza essas mudanças, além de nos permitir versionamento da documentação, onde mitiga possíveis erros e desatualização de regras, requerimentos e informações relevantes para o projeto.”

A análise da resposta fornecida para a Questão 6 (Q6) do GQM revela como a entrega contínua de documentação foi abordada considerando sua acuidade no ciclo do projeto. A documentação contínua é vista pelo líder como uma abordagem estratégica, possibilitando sensibilização a mudanças e versionamento eficaz.

- **Pergunta 7:** Quais os benefícios e dificuldades que você constatou sobre a forma de entrega (disponibilização, acesso e visualização) da documentação na plataforma?

“A plataforma é de fácil interação e manuseio, além de seguir padrões de segurança global e fácil gerenciamento em termos de configuração e disponibilização de todos os recursos para os membros da equipe. Podendo destacar que os ganhos são relevantes no projeto em todas as fases. Além de centralizar em único repositório de forma segura e fácil acessos aos interessados.”

A análise da resposta fornecida para a Questão 7 (Q7) do GQM revela a forma como a documentação é entregue na plataforma, e foi analisada quanto à interação e segurança. Para o líder, a plataforma destaca-se pela fácil interação, segurança e gerenciamento centralizado, proporcionando ganhos relevantes quanto à disponibilização das informações aos interessados do projeto.

- **Pergunta 8:** A documentação está sendo continuamente entregue? O que poderia melhorar?

“A documentação atende os requerimentos técnicos dos modelos e cenários até então mapeados, dando agilidade a equipe técnica e provocando mudanças consideráveis na cultura dentro da equipe e do cliente. A melhorar entendo que a ferramenta deve ser adaptada a vários cenários de baixa e alta complexidade e dando amplitude e conseqüentemente adapta-la ao longo do tempo. Adicionalmente para líderes técnicos aumenta considerável o número de documentação a serem revisadas, percebo uma oportunidade de agilizar o processo de revisões e governança e automatizar este processo.”

A análise da resposta fornecida para a Questão 8 (Q8) do GQM revela como a continuidade na entrega de documentação foi explorada em termos de atendimento aos requisitos técnicos. Segundo a avaliação final do líder, a documentação atende aos requisitos, gerando agilidade e desafios percebidos em adaptar a ferramenta a diferentes cenários.

As respostas coletadas apontam para diversos benefícios decorrentes da implementação das diretrizes com base no *DevDocOps* na plataforma. Destacam-se a melhoria na agilidade da documentação, proporcionando uma rápida geração de artefatos detalhados e a conseqüente aceleração do desenvolvimento. Também, o aumento na quantidade de documentação entregue durante as *sprints* evidencia

uma maior cobertura e detalhamento técnico, favorecendo a equipe no cumprimento eficaz das demandas.

A padronização da documentação foi apontada como uma área de significativa melhoria, refletindo em documentos mais claros, objetivos e visualmente apresentados de maneira consistente e padronizada. Além disso, a comparação entre o processo anterior e o atual revela ganhos significativos em termos de tempo, padronização e detalhamento técnico, proporcionando visibilidade à equipe e liderança sobre o cumprimento das documentações.

A obrigatoriedade da documentação contínua é percebida como uma prática essencial, trazendo benefícios como a vitalidade da documentação ao longo do ciclo do projeto e o versionamento que mitiga possíveis erros e desatualizações. A plataforma, por sua vez, é elogiada por sua facilidade de interação, padrões de segurança e eficiente gerenciamento, centralizando recursos de forma segura e acessível para a equipe.

Entretanto, a observação de oportunidades de melhoria incluem a necessidade de adaptação contínua da ferramenta para diferentes cenários e a automação de processos de revisão e governança para otimizar a eficácia da documentação contínua. Essas considerações destacam a importância da flexibilidade da plataforma e a automação de tarefas para garantir a eficiência e adaptabilidade do processo no decorrer do tempo.

3.4.3 Métricas e Análise de Resultados

Nesta seção será apresentados os detalhes das métricas que foram definidas e monitoradas ao longo do estudo experimental. As métricas desempenham um papel fundamental na avaliação do impacto do *DevDocOps* nas atividades de desenvolvimento de software e na produção de documentação. Será fornecida uma análise aprofundada dos resultados obtidos com base nessas métricas, destacando os principais resultados e conclusões que emergiram durante o estudo. Essa análise nos permitirá compreender melhor os benefícios e desafios do *DevDocOps* no contexto de desenvolvimento ágil de engenharia de dados.

Este projeto tem como foco principal a integração e otimização do processo de documentação de engenharia de dados no contexto da metodologia *DevDocOps*. As metas definidas na Tabela 3 visam melhorar tanto a eficiência quanto a qualidade da documentação gerada, garantindo que ela esteja alinhada com as práticas ágeis e os requisitos de entrega contínua, sendo representadas na visão geral abaixo:

- **Agilidade:** A meta é otimizar o tempo necessário para produzir documentação de qualidade, integrando-a de maneira eficiente no ciclo de vida do desenvolvimento de engenharia de dados.
- **Qualidade:** A meta é estabelecer padrões para a documentação, assegurando que ela seja precisa, detalhada e útil para todas as partes interessadas, incluindo desenvolvedores, gerentes de projeto e clientes.
- **Entrega Contínua de Documentação:** Assegurar que a documentação esteja sempre atualizada e em sincronia com as versões mais recentes do código, facilitando a consistência entre as versões.
- **Acesso à Documentação:** Garantir que a documentação esteja facilmente acessível para todas as partes interessadas, com informações claras e atualizadas.

A implementação do *DevDocOps*, uma abordagem destinada a otimizar o processo de documentação em um ambiente de desenvolvimento ágil, trouxe consigo a promessa de melhorias significativas a respeito a entrega de documentação contínua. Para avaliar com precisão o impacto da mudança proposta pelas abordagens que esta pesquisa definiu, foram definidas uma série de métricas e indicadores que foram monitorados ao longo do estudo experimental. Agora, neste estágio da análise, os resultados obtidos serão comparados com as metas estabelecidas anteriormente.

As métricas definidas no experimento foram projetadas para capturar uma variedade de aspectos relacionados à produção de documentação, qualidade, eficiência e impacto do *DevDocOps* nas atividades de desenvolvimento. Essas métricas incluem o tempo gasto na documentação, a quantidade de documentação produzida,

a padronização da documentação, a facilidade de acesso à documentação e muito mais.

Esta análise tem como objetivo fornecer uma visão aprofundada do desempenho do *DevDocOps* em relação a seis metas específicas:

- **Eficiência na Produção de Documentação:** Avalia se o novo processo acelerou a produção de documentação, reduzindo o tempo gasto nessa atividade.
- **Aderência a Padrões:** Verifica se a documentação gerada está em conformidade com os padrões estabelecidos, identificando desvios que possam exigir correção.
- **Identificação de Gargalos e Pontos de Melhoria:** Analisa os obstáculos remanescentes no processo e proporemos melhorias para otimização.
- **Redução de Erros e Retrabalhos:** Medição do impacto na qualidade do documento entregue, analisando a quantidade de correções necessárias após a geração automatizada.
- **Avaliação da Rastreabilidade e Transparência:** Facilidade do rastreamento e a colaboração, avaliando a transparência alcançada pela documentação.
- **Satisfação dos Desenvolvedores e Líder:** Coleta de *feedbacks* e avaliações para compreender o nível de satisfação e identificar áreas de melhoria.

No contexto das perspectivas dos desenvolvedores, as métricas estabelecidas anteriormente tiveram como objetivo medir diferentes aspectos que afetam diretamente o trabalho dos desenvolvedores e sua interação com o novo processo. A seguinte análise de como cada uma dessas metas foi atingida foi obtida com base nos insumos obtidos pelos desenvolvedores por meio do questionário e *feedback*.

- **Eficiência na Produção de Documentação e Impacto do Processo *DevDocOps* no Tempo de Documentação:**

Pode-se observar que os resultados obtidos pelas duas métricas possuem forte correlação, considerando que a implementação do *DevDocOps* resultou em uma redução significativa no tempo necessário para produzir documentação técnica. Isso foi comprovado a partir do registro dos desenvolvedores que relataram que o processo automatizado agilizou a geração de documentação, permitindo que eles se concentrassem em outras tarefas de desenvolvimento.

- **Precisão da Documentação e Conformidade entre Documentação e Código em Produção:**

A métrica de conformidade revelou que a documentação gerada pelo *DevDocOps* estava alinhada com o código em produção, enquanto a métrica de precisão possui relação com o nível de detalhamento da documentação técnica entregue. Os resultado de cumprimento das duas métricas deve-se ao destacamento dos desenvolvedores a respeito da precisão da documentação, indicando que ela refletia com exatidão os modelos de dados e as funcionalidades implementadas.

- **Tempo de Resposta para Acesso à Documentação:** A métrica Tempo de Resposta para Acesso à Documentação também pode ser descrita por Nível de Interesse na Documentação, descrevendo sobre a forma e facilidade de acesso à documentação criada automaticamente na plataforma. Como cumprindo das métricas, os desenvolvedores expressaram satisfação com a rapidez e facilidade de acesso à documentação, demonstrando que o *DevDocOps* contribuiu para melhorar a eficiência no acesso às informações necessárias para o desenvolvimento.

Essas conclusões sugerem que, na perspectiva dos desenvolvedores, o *DevDocOps* teve um impacto positivo na eficiência da produção de documentação, na precisão dos documentos e no acesso ágil às informações técnicas. Os resultados indicam que as metas estabelecidas para essa perspectiva foram alcançadas com sucesso, melhorando a eficácia geral do processo de documentação.

Na perspectiva do líder, as métricas estabelecidas tiveram como objetivo avaliar como o *DevDocOps* impactou o trabalho e as responsabilidades do líder. As

respostas obtidas no questionário preenchidos pelo líder forneceram os resultados necessários para a avaliação sobre como o processo influenciou o tempo de documentação, a aderência a padrões, a precisão da documentação e o nível de interesse na documentação técnica.

- **Impacto do Processo *DevDocOps* no Tempo de Documentação:**

As respostas do líder, em complemento aos resultados dos testes dos desenvolvedores, indicaram uma melhoria significativa no tempo de documentação com a implementação do processo *DevDocOps*. A meta foi alcançada com base na análise do processo simplificado e automatizado que foi aplicado, que permitiu que a equipe de desenvolvimento produzisse documentação de forma mais rápida e eficiente, liberando tempo para se concentrar em outras atividades críticas do projeto. Isso contribuiu para uma redução no tempo geral dedicado à documentação técnica.

- **Aderência a Padrões:**

O líder destacou que o *DevDocOps* melhorou a aderência da equipe aos padrões de documentação estabelecidos. A automação e a padronização inerentes ao processo garantiram que as documentações fossem consistentes e seguissem os padrões predefinidos. Embora houvessem melhorias sugeridas pelos desenvolvedores, o resultado obtido no experimento demonstrou uma documentação mais organizada, clara e fácil de revisar, o que é fundamental para manter a qualidade e a consistência nas informações técnicas, assegurando o cumprimento da métrica.

- **Precisão da Documentação:**

A precisão da documentação foi outra área beneficiada pela implementação do *DevDocOps*. O líder observou que o processo automatizado reduziu erros humanos e inconsistências nas documentações técnicas. O cumprimento da métrica se deu a partir do resultado da documentação ter se tornado mais confiável, refletindo com maior precisão o estado atual do código em produção.

- **Nível de Interesse na Documentação:**

Um dos aspectos notáveis foi o aumento do nível de interesse da equipe na documentação técnica. O líder percebeu que, com o *DevDocOps*, a documentação se tornou mais acessível e valiosa para a equipe de desenvolvimento. Isso incentivou os membros da equipe a se envolverem mais com a documentação, tornando-se mais autônomos em relação à sua utilização. O aumento do interesse na documentação pode ser atribuído à facilidade de acesso, qualidade e utilidade da documentação produzida pelo processo *DevDocOps*, tornando a métrica então cumprida.

Os resultados obtidos pelas métricas confirmam a eficácia do processo em atender às metas estabelecidas e beneficiar a perspectiva do líder no contexto de desenvolvimento de software.

3.4.4 Análise do Feedback dos Voluntários sobre a Utilização da Plataforma

Após a conclusão do experimento com a plataforma proposta, foi realizada uma reunião de finalização para coletar *feedback* dos voluntários envolvidos. Apesar de repetir algumas das mesmas percepções já relatadas nos questionários individuais, esta sessão foi crucial para avaliar a eficácia da plataforma, bem como identificar áreas de melhoria. Os desenvolvedores compartilharam suas experiências e percepções, abrangendo aspectos críticos como entrega contínua, qualidade da documentação, obrigatoriedade de entrega, acesso à documentação, uso cotidiano da plataforma e sua intuitividade.

Com base no entendimento das experiências e percepções compartilhadas pelos desenvolvedores, foi possível observar os aspectos críticos como entrega contínua, qualidade da documentação, obrigatoriedade de entrega, acesso à documentação, uso cotidiano da plataforma e sua intuitividade. Abaixo são destacados os insumos coletados:

- Entrega Contínua

Os voluntários observaram que o processo de entrega contínua, seguindo o padrão de um artefato único, facilitou significativamente o acesso à docu-

mentação. Foi destacado que no processo anterior, haviam casos em que a documentação não podia ser finalizada na *sprint* atual, geralmente se devendo a restrições de tempo e prioridades de entrega, destacando a melhoria da eficiência da documentação no processo automatizado, devido ao ganho de tempo em realizar tais ações.

- Qualidade da Documentação

A qualidade da documentação, especialmente dos dicionários de dados, foi elogiada, pois não necessitou de ajustes e apresentou uma qualidade de uso satisfatória. No entanto, houve sugestões para melhorar a visualização das hierarquias das *Calculation Views*, indicando a necessidade de aprimoramentos na apresentação de objetos complexos.

- Obrigatoriedade de Entrega com Documentação

Os desenvolvedores relataram que a obrigatoriedade de entregar a tarefa juntamente com a documentação não afetou adversamente os prazos de entrega. Pelo contrário, a facilidade de uso e a rapidez da plataforma foram vistas como benefícios significativos.

- Acesso à Documentação

Quanto ao acesso à documentação, os voluntários apontaram a simplicidade e acessibilidade da plataforma. Embora o conhecimento prévio da plataforma fosse necessário, uma vez familiarizados, os usuários acharam o sistema intuitivo e fácil de navegar.

- Integração e Desenvolvimento Contínuo (CI/CD)

Os *feedbacks* indicaram que não havia limitações significativas na prestabilidade de tarefas dentro das diretrizes propostas. Os desenvolvedores encontraram aplicabilidade em todos os cenários de desenvolvimento nos quais participaram, sugerindo uma adaptabilidade robusta dos *scripts* automatizados a diversos contextos.

- Melhorias Sugeridas

Embora o processo tenha sido geralmente bem recebido, os voluntários sugeriram algumas melhorias. Por exemplo, foi recomendado um acompanhamento inicial para facilitar a compreensão da plataforma. Além disso, sugeriu-se a automação da exportação de configurações de tarefas para evitar erros manuais, melhorando a eficiência do processo.

- Nível Técnico e Eficiência de Tempo

Os desenvolvedores destacaram a simplicidade e a intuitividade da plataforma, notando que ela não requer conhecimentos técnicos aprofundados. A capacidade de carregar um arquivo e ver o resultado final sem a necessidade de criar *branches* secundárias foi sugerida. Quanto ao tempo de uso, os desenvolvedores relataram uma média de 10 minutos para concluir suas tarefas, indicando uma aprendizagem rápida e eficiência de tempo.

Por meio de uma sessão individual com o líder, foi possível obter o *feedback* sobre a implementação e o impacto da plataforma em questões de agilidade, qualidade, continuidade e padronização da documentação. Abaixo são destacados os insumos coletados:

- Especificação Funcional e Técnica

O líder destacou a importância da especificação funcional e técnica, enfatizando que este processo envolve a documentação clara e precisa de todos os requisitos técnicos do cliente. A especificação técnica cobre os procedimentos técnicos detalhados relacionados ao desenvolvimento de produtos, garantindo que todas as informações essenciais sejam documentadas de maneira abrangente. Esta abordagem “viva” da documentação, que se adapta e evolui ao longo do ciclo do projeto, foi reconhecida como um benefício crucial, proporcionando visibilidade técnica e facilitando a tomada de decisão. Segundo o líder, em comparação com métodos tradicionais, o novo processo de documentação atendeu melhor às necessidades técnicas, proporcionando ganhos em tempo, padronização e detalhamento técnico.

- Agilidade da Documentação

O líder destacou um impacto positivo significativo na agilidade da documentação. A implementação do processo *DevDocOps* proporcionou uma documentação ágil e confiável, melhorando a visibilidade e o detalhamento técnico. Isso não só facilitou a transferência de conhecimento dentro da equipe, mas também fornece agilidade no treinamento de novos membros.

- Benefícios da Documentação

O líder apontou vários benefícios da especificação detalhada, incluindo a promoção de um entendimento comum entre todas as partes interessadas, servindo como um ponto de referência confiável durante o desenvolvimento e identificando lacunas nos requisitos em estágios iniciais. Este processo contribuiu para a redução de custos e tempo de desenvolvimento, evitando retrabalho e mal-entendidos.

- Desenvolvimento de Estórias

Quanto à agilidade no desenvolvimento de estórias, o líder observou uma melhoria na geração rápida de artefatos, permitindo que a equipe técnica se concentrasse em outros entregáveis. No entanto, foi apontada a necessidade de um monitoramento mais rigoroso da entrega realizada pelos desenvolvedores e de uma governança aumentada nas revisões dos artefatos.

- Metas do Projeto

Relacionando com as metas estabelecidas, o líder reconheceu a eficácia do processo em agilizar a produção da documentação, garantir sua qualidade e padronização, e facilitar a entrega contínua. Ele enfatizou que a documentação atende às necessidades técnicas, cumprindo um papel estratégico no projeto e incentivando uma cultura de documentação eficaz.

A análise detalhada do *feedback* do líder, juntamente com as informações adicionais fornecidas, reforça a percepção de que a plataforma implementada atendeu e superou as expectativas em termos de agilidade, qualidade, padronização e continuidade da documentação.

3.4.5 Conclusão do Experimento

A ênfase na especificação técnica, os benefícios destacados e a integração com os modelos de dados implementados no SAP são indicativos do sucesso do projeto. As sugestões de melhoria, como a adaptação da ferramenta a diferentes cenários e a automatização do processo de revisão, são diretrizes valiosas para futuras iterações e desenvolvimentos da plataforma. Essa abordagem holística, que abrange desde a especificação técnica até a implementação prática, é fundamental para garantir que a documentação não apenas cumpra os requisitos técnicos, mas também se integre efetivamente aos processos e à cultura da equipe e dos *stakeholders* envolvidos. A liderança vê a documentação contínua como um investimento estratégico, essencial para a evolução e o sucesso do projeto.

4 Diretrizes para a Documentação de Engenharia de Dados em uma abordagem DevDocOps

Neste capítulo, será apresentado uma proposta que emerge da análise abrangente da literatura, bem como dos resultados obtidos no experimento realizado sobre a implementação da abordagem *DevDocOps* no contexto de Engenharia de Dados. A abordagem *DevDocOps* visa otimizar a produção contínua de documentação técnica, desempenhando um papel fundamental na gestão e comunicação eficaz de informações técnicas em um ambiente de desenvolvimento ágil.

Com base na pesquisa apresentada e nas percepções coletadas dos desenvolvedores e líder, esta pesquisa também visou propor a definição de Diretrizes para a Documentação de Engenharia de Dados, que buscam promover a qualidade, a consistência e a eficiência na criação, revisão e disponibilização de documentações técnicas nesse contexto específico. Essas diretrizes representam uma contribuição valiosa para aprimorar a prática da documentação de Engenharia de Dados no âmbito do *DevDocOps* e podem servir como um guia sólido para equipes de desenvolvimento em busca de excelência na gestão de informações técnicas.

A abordagem *DevDocOps*, como aplicada no experimento, é uma metodologia que combina os princípios do Desenvolvimento de Software (Dev) com a documentação técnica (Doc) e as práticas de Operações (Ops) em um ciclo de desenvolvimento integrado e contínuo.

A proposta de diretrizes para a aplicação do *DevDocOps* no setor de engenharia de dados vai além da simples promoção da documentação automatizada, abrangendo também a integração eficaz na gestão do desenvolvimento. Essa integração cobre a organização de atividades e entregas por parte dos desenvolvedores, garantindo que tanto a gestão do fluxo de trabalho quanto o desenvolvimento estejam em sintonia com o processo.

O desenvolvimento dessas diretrizes resultou na criação de ferramentas técnicas altamente adaptáveis, que facilitam a implementação dessa abordagem em todo o ciclo de engenharia de dados. Estas incluem a sincronização contínua da documentação com o código-fonte e a automação dos processos de criação e revisão de documentação, além de estabelecer padrões de qualidade e acessibilidade.

Ao implementar esta metodologia na engenharia de dados, é crucial ressaltar a combinação das práticas ágeis com uma documentação e integração contínuas eficientes, utilizando a plataforma *Azure DevOps*. Este enfoque adapta as metodologias ágeis às exigências específicas do setor, enfatizando o uso de pipelines de CI/CD para a automação e a atualização constante da documentação, assegurando que todas as etapas do processo sejam gerenciadas de forma coesa.

As diretrizes propostas apresentam duas vertentes distintas de responsabilidade. Uma delas é atribuída aos desenvolvedores, que são responsáveis pela criação, manutenção e atualização da documentação à medida que avançam nos processos de desenvolvimento. A outra vertente recai sobre o líder, que tem a responsabilidade de supervisionar e assegurar que as práticas de documentação e os fluxos de trabalho de integração contínua estejam alinhados com os objetivos gerais do projeto e as práticas ágeis da organização.

A seguir, as diretrizes específicas são descritas detalhadamente, abordando os elementos cruciais para a eficácia do *DevDocOps* na engenharia de dados. Estas diretrizes são projetadas para orientar tanto os desenvolvedores quanto os líderes em suas respectivas responsabilidades. Para os desenvolvedores, enfocam a importância da documentação clara e concisa do código e das mudanças, assegurando a manutenibilidade e compreensão futuras. Para os líderes, ressaltam a necessidade de revisar e validar a documentação para garantir que ela atenda a todas as exigências técnicas e padrões do projeto. Ambos os papéis são estruturados de maneira a complementar um ao outro, maximizando a eficiência e a qualidade do produto final entregue no ambiente de engenharia de dados.

1. Liderança Orientada em DevDocOps: Personalizando a Documentação e Gestão em Projetos de Engenharia de Dados

Como principal aplicador da prática, cabe ao líder avaliar e personalizar o *template* de documentação definido para cada projeto e cliente. Essa tarefa inclui estabelecer regras específicas para a documentação, como formatação, profundidade das informações e identificação das necessidades técnicas. Além disso, o líder é responsável por garantir que as diretrizes sejam adaptadas às características únicas de cada equipe. O *script* genérico fornecido no estudo experimental oferece uma base que pode ser aprimorada e personalizada de acordo com as exigências individuais de cada projeto, permitindo que o líder oriente de forma eficaz a equipe na implementação de práticas de documentação que sejam ao mesmo tempo ágeis e abrangentes, adequando-se às demandas específicas do ambiente de engenharia de dados.

2. Gestão de *Backlog* para Rastreamento Eficiente

Para assegurar um rastreamento eficaz dos objetos desenvolvidos, é crucial que o líder estruture o *backlog* de forma a agrupar atividades relacionadas a uma mesma alteração de objeto durante a *sprint*. Embora as tarefas devam refletir o menor fragmento de software possível, seguindo os princípios ágeis, na engenharia de dados, é essencial considerar o funcionamento adequado do objeto devido às suas dependências e impactos no ambiente produtivo. Assim, as alterações no objeto serão refletidas com mais clareza no documento final, garantindo um detalhamento mais preciso e visível das mudanças efetuadas.

3. Garantia de Qualidade e Segurança na Entrega de Projetos

No fluxo proposto, é essencial que cada entrega inclua uma revisão detalhada dos objetos desenvolvidos pelos programadores, visto que o *deploy* agora requer a aprovação do líder antes de ser efetuado na *branch* principal do repositório. Esta abordagem eleva a segurança da solução entregue, exigindo, contudo, um controle e acompanhamento mais rigoroso e direcionado em cada fase de entrega, reforçando a qualidade e a confiabilidade do produto final.

4. Gestão Segura de Alterações e Implantação no DevOps

O desenvolvedor é responsável por gerenciar as alterações de forma a prevenir *deploys* diretos na *branch* principal do repositório. Esta prática eleva a segurança das implantações, minimizando riscos e impactos no ambiente de

produção. Além disso, facilita a ativação automática do pipeline de documentação contínua, que é disparado após a aprovação do código pelo líder. Este método assegura uma integração segura e eficaz das alterações no código, promovendo a estabilidade e a confiabilidade do sistema.

5. Centralização da Documentação em *DevDocOps*: Uma Prática de Acesso e Segurança

No contexto da "Garantia de Qualidade e Segurança na Entrega", o fluxo automatizado para geração de documentação após validação do código implica que a documentação criada seja integrada ao contexto principal do projeto, com opção de download. É vital que o líder promova a prática de manter a documentação centralizada em um único local, apesar da possibilidade de transferência externa dos documentos. Isso facilita o acesso a registros históricos e atuais, consolidando uma melhor prática no *DevDocOps* para garantir a acessibilidade e a segurança da informação.

6. Afinando a Documentação Automatizada

Embora o processo de geração de documentação seja automatizado, garantindo um padrão uniforme para todos os objetos desenvolvidos, é crucial considerar a possibilidade de que, em certos casos, a documentação possa requerer ajustes adicionais, especialmente para objetos de maior complexidade. O desenvolvedor é responsável por completar a documentação com uma revisão detalhada, enquanto o líder tem a tarefa de verificar se o documento final satisfaz as exigências técnicas e as necessidades do projeto.

7. Conclusão e Comunicação Efetiva no Ciclo de Vida das Tarefas de Documentação:

A comunicação é essencial e deve ser refletida nas atualizações das tarefas definidas no quadro de gestão de projetos. O registro de todas as entregas de documentação deve ser claramente marcado nas tarefas correspondentes. O ciclo de vida de uma tarefa se conclui com a entrega da documentação associada, garantindo que cada fase do processo esteja devidamente documentada e comunicada, desde o início do desenvolvimento até a sua conclusão.

8. Personalização e Eficiência no *DevDocOps*: Pipelines Dedicados para Documentação de Objetos

A documentação deve ser considerada um elemento orgânico no ciclo de vida do projeto, sujeita a personalização conforme as necessidades específicas do *DevDocOps*. Recomenda-se a implementação de pipelines distintos para cada objeto de dados, o que não apenas acelera a execução, mas também permite abordar especificidades técnicas individuais de cada elemento. Essa abordagem modular e flexível assegura uma gestão mais eficiente e adaptável da documentação, alinhada com os requisitos técnicos e as características de cada objeto dentro do projeto.

A conclusão deste projeto abrange a efetiva implementação do *DevDocOps* na engenharia de dados, destacando a integração de práticas ágeis, gestão eficiente da documentação e integração contínua. A pesquisa e o experimento demonstraram como a abordagem *DevDocOps* otimiza a produção contínua de documentação técnica, essencial para a comunicação eficaz em ambientes de desenvolvimento ágil. Esta metodologia, aplicada ao contexto da engenharia de dados, enfatizou a importância da liderança na avaliação e personalização da documentação, bem como na gestão de atividades e tarefas.

As diretrizes propostas visam promover a qualidade, a consistência e a eficiência na criação e revisão de documentações técnicas, refletindo a necessidade de adaptabilidade e personalização no *DeDevDocOps*. O líder desempenha um papel crucial no sucesso deste processo, garantindo que a documentação atenda às necessidades técnicas e padrões estabelecidos, enquanto os desenvolvedores contribuem com a documentação detalhada e clara das funcionalidades.

Por fim, a abordagem *DevDocOps*, enriquecida com pipelines personalizados para diferentes objetos, demonstrou ser uma estratégia adaptável e eficaz para melhorar a gestão de informações técnicas e a qualidade da documentação em projetos de engenharia de dados. A implementação dessas diretrizes assegura um desenvolvimento de dados mais estruturado, organizado e de alta qualidade, estabelecendo um novo padrão para projetos nesse âmbito.

5 Conclusão

Esta pesquisa explorou a integração da documentação no processo de desenvolvimento ágil de software, com foco na abordagem *DevDocOps*. O estudo apresentou um panorama detalhado das práticas atuais e desafios enfrentados na documentação de software, particularmente na engenharia de dados, e propôs uma série de diretrizes para otimizar este processo.

5.1 Principais Descobertas

As descobertas desta pesquisa promovem resultados no âmbito da integração da documentação no processo de desenvolvimento ágil de software, com um enfoque particular na abordagem *DevDocOps*. Uma das principais revelações foi a eficácia da automatização da documentação em melhorar tanto a eficiência quanto a precisão da documentação em ambientes de desenvolvimento ágeis. A abordagem *DevDocOps* provou ser um modelo promissor para resolver problemas de sincronização entre código e documentação, contribuindo para um ciclo de desenvolvimento mais ágil e preciso.

Além disso, uma descoberta notável deste estudo foi a constatação de que a correlação entre as metas específicas deste projeto e os resultados práticos ainda não foi amplamente investigada na literatura existente. Apesar do crescente interesse em práticas ágeis e na integração da documentação no ciclo de desenvolvimento de software, pouco se tem estudado sobre como as metas específicas de projetos como este se alinham e impactam os resultados finais em ambientes de engenharia de dados.

Essa lacuna na literatura sugere uma oportunidade significativa para futuras pesquisas. A correlação entre as metas de automatizar a documentação, melhorar a qualidade da mesma, e promover a entrega contínua, com os resultados práticos obtidos em projetos de engenharia de dados, permanece um campo fértil para investigação. Compreender como essas metas interagem e influenciam os resultados

pode fornecer insights valiosos para otimizar ainda mais as práticas de *DevDocOps* e melhorar o processo de desenvolvimento ágil como um todo.

A constatação de que existe uma necessidade de mais pesquisas para explorar essa correlação ressalta a importância deste projeto de pesquisa. As descobertas aqui apresentadas não apenas contribuem para o corpo de conhecimento existente, mas também abrem caminhos para novas investigações que podem aprofundar nossa compreensão e eficácia do *DevDocOps* em ambientes de engenharia de dados.

5.2 Implicações Práticas

5.2.1 Versatilidade e Adaptabilidade do Modelo

Uma das principais implicações deste estudo é a versatilidade do modelo *DevDocOps* proposto. Por ser um modelo genérico, ele pode ser adaptado a uma variedade de contextos práticos dentro de uma organização. Essa flexibilidade é crucial em um ambiente empresarial dinâmico, onde as necessidades e os processos podem variar significativamente entre diferentes departamentos, projetos e equipes. A capacidade de adaptar o modelo *DevDocOps* a esses diferentes cenários significa que organizações de todos os tipos e tamanhos podem implementar as práticas recomendadas, ajustando-as conforme necessário para atender às suas necessidades específicas.

Essa adaptabilidade também permite que a abordagem *DevDocOps* seja aplicada não apenas em projetos de engenharia de dados, mas também em outras áreas como desenvolvimento de software, gestão de projetos, e até em departamentos não técnicos que podem se beneficiar de uma documentação mais ágil e eficiente. Isso amplia significativamente o escopo de aplicação do modelo, tornando-o uma solução valiosa para uma ampla gama de desafios de documentação em toda a organização.

5.2.2 Otimização de Tempo e Eficiência de Tarefas

Além da versatilidade do modelo, as descobertas também enfatizam a otimização de tempo e a eficiência das tarefas da equipe. A implementação das

diretrizes *DevDocOps* levou a uma redução significativa no tempo necessário para a documentação técnica e a uma melhoria na qualidade geral da documentação. Essa otimização do tempo, aliada à automatização de várias tarefas de documentação, permite que as equipes se concentrem em aspectos mais críticos do desenvolvimento, melhorando a eficiência e a produtividade.

5.2.3 Conclusão das Implicações Práticas

Considerando os pontos apresentados, as implicações práticas deste projeto de pesquisa representam um avanço em direção à melhoria da geração de artefatos de conhecimento em metodologias ágeis. A facilidade de aplicabilidade, a otimização do tempo e a eficiência das tarefas, juntamente com a versatilidade e adaptabilidade do modelo *DevDocOps*, destacam-no como uma abordagem inovadora e valiosa para a modernização dos processos de desenvolvimento de software e dados. Estas descobertas oferecem um roteiro claro para a implementação bem-sucedida do *DevDocOps* e abrem novas possibilidades para melhorias contínuas em uma ampla gama de contextos dentro das organizações.

5.3 Recomendações para Futuras Pesquisas

Este estudo forneceu uma base sólida para a integração da documentação no desenvolvimento ágil de software, especialmente com a abordagem *DevDocOps*. No entanto, existem áreas que ainda requerem investigação mais aprofundada. Duas áreas promissoras para futuras pesquisas incluem a expansão da automatização para mais tipos de objetos de engenharia de dados e a avaliação da eficácia dos *templates* padronizados na melhoria da documentação.

5.3.1 Avaliação Investigativa de Melhoria dos *Templates* Padronizados de Documentação

Os *templates* de documentação desempenham um papel crucial na qualidade da documentação, na manutenção da consistência e na facilitação de sua compreensão. Uma área que merece atenção futura é a avaliação da eficácia dos *templates*

padronizados na exibição da documentação. Pesquisas futuras poderiam explorar como diferentes formatos e estruturas de *templates* afetam a usabilidade, a clareza e a eficácia da documentação. Isso incluiria não apenas a análise de *templates* existentes, mas também o desenvolvimento e teste de novos modelos que podem ser mais adequados para certos tipos de projetos ou ambientes de engenharia de dados. Uma avaliação investigativa detalhada poderia fornecer orientações valiosas para a criação de documentação mais intuitiva e informativa, beneficiando tanto os desenvolvedores quanto os usuários finais.

5.3.2 Conclusão Final das Recomendações

Através dessas investigações futuras, podemos avançar significativamente na maneira como abordamos a documentação no desenvolvimento de software, especialmente em ambientes complexos de engenharia de dados. As pesquisas recomendadas têm o potencial de ampliar nossa compreensão sobre a eficiência da documentação automatizada e a eficácia de diferentes estratégias de apresentação, contribuindo para a evolução contínua das práticas de *DevDocOps* e, por extensão, do desenvolvimento ágil de software como um todo.

5.4 Conclusão Final

Este trabalho destacou a importância e a viabilidade da abordagem *DevDocOps* no desenvolvimento ágil de software, com um foco especial na engenharia de dados. As diretrizes propostas, fruto de uma análise aprofundada e de um estudo experimental, oferecem um caminho promissor para a melhoria da qualidade e eficiência da documentação em projetos de software. A esperança é que este estudo sirva como um ponto de partida para futuras pesquisas e práticas no campo do desenvolvimento ágil de engenharia de dados, especialmente no que diz respeito à integração eficiente da documentação no ciclo de vida do desenvolvimento de software.

Referências

- AGHAJANI, E. et al. Software documentation issues unveiled. In: . [S.l.]: IEEE Computer Society, 2019. v. 2019-May, p. 1199–1210. ISBN 9781728108698. ISSN 02705257. Citado na página 42.
- ATWAL, H. The dataops factory. In: _____. *Practical DataOps: Delivering Agile Data Science at Scale*. Berkeley, CA: Apress, 2020. p. 249–266. ISBN 978-1-4842-5104-1. Disponível em: <https://doi.org/10.1007/978-1-4842-5104-1_10>. Citado na página 35.
- BASS, L. The software architect and devops. *IEEE Software*, v. 35, p. 8–10, 2017. ISSN 07407459. Citado na página 27.
- BERHOUMA, H. A generic model for software documentation and its application in embedded systems developed with scrum. In: . [S.l.]: Association for Computing Machinery, 2020. p. 33–36. ISBN 9781450377218. Citado na página 34.
- CAPIZZI, A.; DISTEFANO, S.; MAZZARA, M. From devops to devdataops: Data management in devops processes. In: BRUEL, J.-M.; MAZZARA, M.; MEYER, B. (Ed.). *Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment*. Cham: Springer International Publishing, 2020. p. 52–62. ISBN 978-3-030-39306-9. Citado na página 35.
- FÄRBER, F. et al. Sap hana database. *ACM SIGMOD Record*, v. 40, p. 45–51, 1 2012. ISSN 0163-5808. Citado 3 vezes nas páginas 39, 41 e 53.
- GOKARNA, M.; SINGH, R. *DevOps: A historical review and future works*. 2020. Citado na página 33.
- JAIN, R.; SUMAN, U. Root causes and reduction techniques for rework in global software development. *International Journal of Computer Applications*, Foundation of Computer Science, v. 183, p. 40–44, 12 2021. Citado na página 53.
- LEITE, L. et al. A survey of devops concepts and challenges. *arXiv*, v. 52, 2019. ISSN 23318422. Citado 4 vezes nas páginas 33, 34, 37 e 52.
- MIZRAHI, E. *How to Automate Data Documentation*. 2023. Citado na página 38.
- MUÑOZ, M.; RODRÍGUEZ, M. N. A guidance to implement or reinforce a devops approach in organizations: A case study. *Journal of Software: Evolution and*

Process, p. e2342, 2021. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/smr.2342>>. Citado na página 34.

PIANINI, D.; NERI, A. Breaking down monoliths with microservices and devops: an industrial experience report. In: *2021 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. [S.l.: s.n.], 2021. p. 505–514. Citado 2 vezes nas páginas 27 e 28.

PONISZEWSKA-MARAÑDA, A.; ZIELISKI, A.; MARAÑDA, W. Towards project documentation in agile software development methods. *Lecture Notes on Data Engineering and Communications Technologies*, Springer Science and Business Media Deutschland GmbH, v. 30, p. 1–18, 2020. ISSN 23674520. Citado 6 vezes nas páginas 28, 33, 37, 38, 43 e 52.

RAMAMOORTHY, C. et al. 1984 iee first international conference on data engineering. In: PRESS, I. C. S. (Ed.). [S.l.]: IEEE, 1984. ISBN 978-0-8186-0533-8. Citado na página 36.

RASHID, S. M. et al. The semantic data dictionary – an approach for describing and annotating data. *Data Intelligence*, MIT Press Journals, v. 2, p. 443–486, 10 2020. ISSN 2641435X. Citado 2 vezes nas páginas 38 e 52.

REIS, J.; HOUSLEY, M.
Fundamentos_{de}engenharia_{de}dados.112023. Citado 2 vezes nas páginas 36 e 37.

RONG, G. et al. Devdocops: Towards automated documentation for devops. *Proceedings - 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice, ICSE-SEIP 2019*, IEEE, p. 243–252, 2019. Citado 7 vezes nas páginas 28, 37, 38, 39, 43, 49 e 52.

RONG, G. et al. Devdocops: Enabling continuous documentation in alignment with devops. *Software: Practice and Experience*, v. 50, n. 3, p. 210–226, 2020. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/spe.2770>>. Citado 2 vezes nas páginas 28 e 43.

ROSSBERG, J. An overview of azure devops azure devops. In: _____. *Agile Project Management with Azure DevOps: Concepts, Templates, and Metrics*. Berkeley, CA: Apress, 2019. p. 37–66. ISBN 978-1-4842-4483-8. Disponível em: <https://doi.org/10.1007/978-1-4842-4483-8_2>. Citado na página 53.

SOLINGEN, R. van; BERGHOUT, E. Integrating goal-oriented measurement in industrial software engineering: industrial experiences with and additions to the goal/question/metric method (gqm). In: *Proceedings Seventh International Software Metrics Symposium*. [S.l.: s.n.], 2001. p. 246–258. Citado na página 54.

SONG, Q. et al. An empirical investigation into the effects of code comments on issue resolution. In: *2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC)*. [S.l.: s.n.], 2020. p. 921–930. Citado na página 43.

SYNKO, A.; PELESHCHYSHYN, A. Software development documenting – documentation types and standards. *Scientific journal of the Ternopil national technical university*, v. 98, p. 120–128, 2020. ISSN 25224433. Citado 4 vezes nas páginas 28, 37, 43 e 52.

TANG, Y.-X. et al. An empirical comparison between tutorials and crowd documentation of application programming interface. *Journal of Computer Science and Technology*, v. 36, p. 856–876, 2021. ISSN 1860-4749. Disponível em: <<https://doi.org/10.1007/s11390-020-0042-0>>. Citado na página 38.

THEUNISSEN, T. Identifying conditions for effective communication with just enough documentation in continuous software development. In: *CAiSE (Doctoral Consortium)*. [S.l.: s.n.], 2020. p. 11–20. Citado 4 vezes nas páginas 37, 42, 48 e 52.

THEUNISSEN, T.; HOPPENBROUWERS, S.; OVERBEEK, S. Approaches for documentation in continuous software development. *Complex Systems Informatics and Modeling Quarterly*, p. 1–27, 10 2022. ISSN 22559922. Citado na página 48.

THEUNISSEN, T.; van Heesch, U.; AVGERIOU, P. A mapping study on documentation in continuous software development. *Information and Software Technology*, v. 142, p. 106733, 2022. ISSN 0950-5849. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S095058492100183X>>. Citado 4 vezes nas páginas 28, 37, 38 e 43.

UNDERWOOD, M. Continuous metadata in continuous integration, stream processing and enterprise dataops. *Data Intelligence*, MIT Press Journals, v. 5, p. 275–288, 12 2023. ISSN 2641435X. Citado na página 34.

VADAVALASA, R. End to end ci/cd pipeline for machine learning. In: . [S.l.: s.n.], 2020. Citado 2 vezes nas páginas 34 e 35.

YARLAGADDA, R. T. Devops and its practices. *International Journal of Creative Research Thoughts (IJCRT)*, v. 9, p. 111–119, 2021. ISSN 2320-2882. Disponível em: <<https://ssrn.com/abstract=3798877>>. Citado na página 34.

ZALEŃSKI, K. Modeling concepts. In: *Data Modeling with SAP BW/4HANA 2.0*. [S.l.]: Springer, 2021. p. 67–96. Citado 2 vezes nas páginas 39 e 40.

ZIMMERMAN, R. P. *A ROLE FOR DATA ANALYSIS IN PRACTICAL REQUIREMENTS DEFINITION*. 1984. Citado na página 36.

APÊNDICE A – Termo de Consentimento Livre e Esclarecido

Termo de Consentimento Livre e Esclarecido

Você está sendo convidado(a) para participar, como voluntário(a), em uma pesquisa científica.

Para confirmar sua participação você precisará ler todo este documento, depois selecionar a opção correspondente no final dele (ACEITO PARTICIPAR ou NÃO ACEITO PARTICIPAR) e deixar um meio para contato contigo, pode ser seu e-mail ou telefone, como você preferir. Este documento se chama TCLE (Termo de Consentimento livre e esclarecido). Nele estão contidas as principais informações sobre o estudo, tais como: objetivos, metodologias, riscos e benefícios, dentre outras informações.

Este TCLE se refere ao projeto de pesquisa "***Diretrizes para a Documentação de Engenharia de Dados em uma abordagem DevDocOps***", cujo objetivo é conceber e avaliar uma abordagem para agilizar a produção e entrega de documentação contínua de software, seguindo a filosofia DevDocOps.

Para ter uma cópia deste TCLE, você poderá imprimi-lo, ou gerar uma cópia em pdf, ou solicitar que seja enviado ao seu e-mail uma versão deste documento.

A pesquisa será realizada por meio de um questionário online, constituído por 11 perguntas, relacionadas a utilização da abordagem DevDocOps para o desenvolvimento e rastreamento de documentação no ciclo de desenvolvimento. Estima-se que você precisará de aproximadamente 10 minutos. A p



Pedir acesso para editar

é determinante para a qualidade da pesquisa.

Os pesquisadores garantem e se comprometem com o sigilo e a confidencialidade de todas as informações fornecidas por você para este estudo. Da mesma forma, o tratamento dos dados coletados seguirá as determinações da Lei Geral de Proteção de Dados (LGPD – Lei 13.709/18).

Para contatar o pesquisador da pesquisa, você poderá encaminhar um e-mail, ligar ou mandar mensagem pelo WhatsApp para ele a qualquer momento:

Nome, celular e e-mail do Pesquisador Responsável: Stephany Mendes Oliveira, (11)95862-6096, stephanymendes@estudante.ufscar.br

CONSENTIMENTO DE PARTICIPAÇÃO

Eu, concordo em participar voluntariamente do presente estudo como participante. O pesquisador me informou sobre tudo o que vai acontecer na pesquisa, o que terei que fazer, inclusive sobre os possíveis riscos e benefícios envolvidos na minha participação.

Fui informado também que devo imprimir ou gerar um pdf do TCLE para ter a minha cópia do TCLE e que posso solicitar uma versão dele via e-mail para os pesquisadores.

[Faça login no Google](#) para salvar o que você já preencheu. [Saiba mais](#)

- ACEITO PARTICIPAR
- NÃO ACEITO PARTICIPAR

Próxima

Limpar formulário



Pedir acesso para editar

Nunca envie senhas pelo Formulários Google.

APÊNDICE B – Questionário de
Utilização do DevDocOps -
Desenvolvedores

Termo de Consentimento Livre e Esclarecido

Você está sendo convidado(a) para participar, como voluntário(a), em uma pesquisa científica.

Para confirmar sua participação você precisará ler todo este documento, depois selecionar a opção correspondente no final dele (ACEITO PARTICIPAR ou NÃO ACEITO PARTICIPAR) e deixar um meio para contato contigo, pode ser seu e-mail ou telefone, como você preferir. Este documento se chama TCLE (Termo de Consentimento livre e esclarecido). Nele estão contidas as principais informações sobre o estudo, tais como: objetivos, metodologias, riscos e benefícios, dentre outras informações.

Este TCLE se refere ao projeto de pesquisa “**Diretrizes para a Documentação de Engenharia de Dados em uma abordagem DevDocOps**”, cujo objetivo é conceber e avaliar uma abordagem para agilizar a produção e entrega de documentação contínua de software, seguindo a filosofia DevDocOps.

Para ter uma cópia deste TCLE, você poderá imprimi-lo, ou gerar uma cópia em pdf, ou solicitar que seja enviado ao seu e-mail uma versão deste documento.

A pesquisa será realizada por meio de um questionário online, constituído por 11 perguntas, relacionadas a utilização da abordagem DevDocOps para o desenvolvimento e rastreo de documentação no ciclo de desenvolvimento. Estima-se que você precisará de aproximadamente 10 minutos. A precisão de suas respostas é determinante para a qualidade da pesquisa.

Os pesquisadores garantem e se comprometem com o sigilo e a confidencialidade de todas as informações fornecidas por você para este estudo. Da mesma forma, o tratamento dos dados coletados seguirá as determinações da Lei Geral de Proteção de

Dados
(LGPD – Lei 13.709/18).

Para contatar o pesquisador da pesquisa, você poderá encaminhar um e-mail, ligar ou mandar mensagem pelo WhatsApp para ele a qualquer momento:

Nome, celular e e-mail
do Pesquisador Responsável: Stephany Mendes Oliveira, (11)95862-6096,
stephanymendes@estudante.ufscar.br

CONSENTIMENTO DE PARTICIPAÇÃO

Eu, concordo em participar voluntariamente do presente estudo como participante. O pesquisador me informou sobre tudo o que vai acontecer na pesquisa, o que terei que fazer, inclusive sobre os possíveis riscos e benefícios envolvidos na minha participação.

Fui informado também que devo imprimir ou gerar um pdf do TCLE para ter a minha cópia do TCLE e que posso solicitar uma versão dele via e-mail para os pesquisadores.

** Indica uma pergunta obrigatória*

1. *Marcar apenas uma oval.*

ACEITO PARTICIPAR
 NÃO ACEITO PARTICIPAR

Informações Pessoais

2. Nome Completo *

3. E-mail *

4. Profissão, cargo ou nível:

Avaliação da Documentação

Esta seção visa abordar a sua visão sobre desenvolvimento, o estado e a entrega de documentação de projetos de dados no seu contexto de testes.

5. 1 - Foi produzida ou atualizada documentação nessa estória ?

Marcar apenas uma oval.

Sim

Não

6. 2 - Quantidade de Tempo Gasto(horas) com a Documentação (tempo para geração automática, revisões)

7. 3 - Se não tivesse sido utilizado o processo DevDocOps, você estimaria ter gastado quanto tempo?

8. 4 - O Objeto em desenvolvimento já possui documentação?

Marcar apenas uma oval.

Sim

Não

9. 4.1 - Caso sua resposta tenha sido "sim" para a questão 4, responda: Você precisou consultar a documentação anterior deste objeto para realizar a tarefa ?

Marcar apenas uma oval.

Sim

Não

10. 4.2 - Caso sua resposta tenha sido "sim" para a questão 4, responda: A documentação anterior estava atualizada com a mesma versão do objeto em produção?

Marcar apenas uma oval.

Sim

Não

11. 4.3 - Caso sua resposta tenha sido "sim" para a questão 4, responda: Foi fácil acessar ou localizar a documentação

Marcar apenas uma oval.

Sim

Não

12. 5 - Foi necessário desenvolver alguma documentação de forma manual ? Se sim, descreva.

Uso da Abordagem DevDocOps

As perguntas desta seção se referem à como foi a sua experiência de utilização da abordagem no dia que foi realizado o teste.

13. 6- Qual a Data do Teste ?

Exemplo: 7 de janeiro de 2019

14. 7 - A documentação precisou de ajustes após ter sido gerada? Se sim, quais:

15. 8 - Quão fácil foi a utilização da plataforma ?

Marcar apenas uma oval.

1 2 3 4 5

Mui Muito Fácil

16. 9 - Houve algum problema na utilização do fluxo DevDocOps?

17. 10 - Com o novo processo, a documentação tornou-se obrigatória. Como você avalia os benefícios e problemas dessa entrega contínua de documentação?

18. 11 - Quais os benefícios e dificuldades que você constatou sobre a forma de entrega (disponibilização, acesso e visualização) da documentação na plataforma ?

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários

APÊNDICE C – Questionário de Utilização do DevDocOps - Líder

Termo de Consentimento Livre e Esclarecido

Você está sendo convidado(a) para participar, como voluntário(a), em uma pesquisa científica.

Para confirmar sua participação você precisará ler todo este documento, depois selecionar a opção correspondente no final dele (ACEITO PARTICIPAR ou NÃO ACEITO PARTICIPAR) e deixar um meio para contato contigo, pode ser seu e-mail ou telefone, como você preferir. Este documento se chama TCLE (Termo de Consentimento livre e esclarecido). Nele estão contidas as principais informações sobre o estudo, tais como: objetivos, metodologias, riscos e benefícios, dentre outras informações.

Este TCLE se refere ao projeto de pesquisa "***Diretrizes para a Documentação de Engenharia de Dados em uma abordagem DevDocOps***", cujo objetivo é conceber e avaliar uma abordagem para agilizar a produção e entrega de documentação contínua de software, seguindo a filosofia DevDocOps.

Para ter uma cópia deste TCLE, você poderá imprimi-lo, ou gerar uma cópia em pdf, ou solicitar que seja enviado ao seu e-mail uma versão deste documento.

A pesquisa será realizada por meio de um questionário online, constituído por 11 perguntas, relacionadas a utilização da abordagem DevDocOps para o desenvolvimento e rastreamento de documentação no ciclo de desenvolvimento. Estima-se que você precisará de aproximadamente 10 minutos. A precisão de suas respostas é determinante para a qualidade da pesquisa.

Os pesquisadores garantem e se comprometem com o sigilo e a confidencialidade de todas as informações fornecidas por você para este estudo. Da mesma forma, o tratamento dos dados coletados seguirá as determinações da Lei Geral de Proteção de Dados (LGPD – Lei 13.709/18).

Para contatar o pesquisador da pesquisa, você poderá encaminhar um e-mail, ligar ou mandar mensagem pelo WhatsApp para ele a qualquer momento:

Nome, celular e e-mail do Pesquisador Responsável: Stephany Mendes Oliveira, (11)95862-6096, stephanymendes@estudante.ufscar.br

CONSENTIMENTO DE PARTICIPAÇÃO

Eu, concordo em participar voluntariamente do presente estudo como participante. O pesquisador me informou sobre tudo o que vai acontecer na pesquisa, o que terei que fazer, inclusive sobre os possíveis riscos e benefícios envolvidos na minha participação.

Fui informado também que devo imprimir ou gerar um pdf do TCLE para ter a minha cópia do TCLE e que posso solicitar uma versão dele via e-mail para os pesquisadores.

* Indica uma pergunta obrigatória

1. *

Marcar apenas uma oval.

ACEITO PARTICIPAR

NÃO ACEITO PARTICIPAR

Informações Pessoais

2. Nome Completo *

3. E-mail *

4. Profissão, cargo ou nível

Avaliação da Documentação e Abordagem DevDocOps

Esta seção visa abordar a sua visão sobre desenvolvimento, o estado e a entrega de documentação de projetos de dados no seu contexto de testes, assim como os resultados obtidos ao utilizar a abordagem DevDocOps.

5. A agilidade da documentação foi impactada positiva ou negativamente com a implantação desse processo ?

6. O que melhorou e o que piorou em relação a agilidade do desenvolvimento das estórias ?

7. A quantidade de documentação entregue na sprint aumentou ?

8. Em termos de padronização da documentação, houve melhoria?

9. Em comparação ao processo de documentação anterior com o novo processo, a documentação entregue cumpre a necessidade técnica ?

10. Com o novo processo, a documentação tornou-se obrigatória. Como você avalia os benefícios e problemas dessa entrega contínua de documentação?

11. Quais os benefícios e dificuldades que você constatou sobre a forma de entrega (disponibilização, acesso e visualização) da documentação na plataforma ?

12. A documentação está sendo continuamente entregue? O que poderia melhorar ?

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários

ANEXO A – Script de Geração de Documentação de Dicionário de Dados

```
1 import pandas as pd
2 import snakemd as snake
3 from datetime import date
4
5 dados = pd.read_csv("./view_columns.csv", delimiter=',')
6
7 infos = pd.read_csv("./config_doc.csv", delimiter=';')
8
9 data = dados.iloc[:,[0,1,2]]
10
11 #Escrita de texto do documento md
12 doc = snake.new_doc("DOCUMENTACAO_TESTE")
13
14 # Filtrando o valor a partir da coluna "VARIABEL"
15 valor_filtrado = infos.loc[infos['VARIABEL'] == 'TITULO', '
    VALOR'].values[0]
16 print(valor_filtrado)
17 arq_title = valor_filtrado
18
19 valor_filtrado = infos.loc[infos['VARIABEL'] == '
    DEV_RESPONSABEL', 'VALOR'].values[0]
20 print(valor_filtrado)
21 arq_dev = valor_filtrado
22
23 valor_filtrado = infos.loc[infos['VARIABEL'] == 'TABELA', '
    VALOR'].values[0]
24 print(valor_filtrado)
25 arq_table = valor_filtrado
26
```

```
27 doc.add_header("Dicionario de Dados - "+ arq_title)
28
29 doc.add_paragraph("Dicionario de Dados do desenvolvimento da
    Tabela "+ arq_table)
30
31     #Recuperando os dados que serao inseridos no documento
32 header_data = data.columns
33 rows = data.to_numpy()
34
35     #Inserindo os dados no arquivo md
36 doc.add_table(
37     header_data,
38     rows
39 )
40
41 data_hora = date.today()
42
43 doc.add_paragraph("Ultima atualizacao"+ str(data_hora))
44
45     #Gerando o arquivo md
46 doc.output_page()
```

Listing A.1 – Código fonte Script Dicionário de Dados

ANEXO B – Script de Geração de Documentação de *Calculation View*

```
1 from lxml import etree
2 import xml.etree.ElementTree as ET
3 import requests
4
5 xml_file = "./Font/calculation_view.xml"
6
7 # Pegando a arvore do xml
8 tree = etree.parse(xml_file)
9
10 # Extraindo a raiz do XML
11 root = tree.getroot()
12 print("Raiz: ", root)
13
14 # Abra um arquivo Markdown para escrita
15 markdown_file = open("output.md", "w")
16
17 # Escreva o titulo do documento
18 markdown_file.write("# Documentacao da View \n\n")
19
20 print("Tabelas fontes da view")
21 filter = "DataSource"
22
23 for i, item in enumerate(root.findall("dataSources/
24     DataSource")):
25     print("{}: {}".format(i, item.attrib["id"]))
26     var_id = "{}: {}".format(i, item.attrib["id"])
27     markdown_file.write("\n")
28     markdown_file.write("{}\n".format(var_id))
```

```
29 nsmmap = {'xsi': 'http://www.w3.org/2001/XMLSchema-instance'}
30
31 for i, item in enumerate(root.findall("calculationViews/
calculationView")):
32     print("\n")
33     print("Calculation Node")
34     markdown_file.write("## Calculation Node\n")
35     print("{}:{}".format(i, item.attrib["id"]), item.
        attrib[f"{{{nsmmap['xsi']}}}type"])
36     var_atrib = "{}:{}".format(i, item.attrib["id"]),
        item.attrib[f"{{{nsmmap['xsi']}}}type"]
37     markdown_file.write("{}\n".format(var_atrib))
38
39     if item.attrib[f"{{{nsmmap['xsi']}}}type"] == "
Calculation:JoinView":
40         print("Ordem do Join: {}".format(item.attrib["
joinOrder"]))
41         var_ordem_j = "{}".format(item.attrib["joinOrder"])
42         markdown_file.write("{}\n".format(var_ordem_j))
43         print("Tipo do Join: {}".format(item.attrib["
joinType"]))
44         var_tipo_j = "{}".format(item.attrib["joinOrder"])
45         markdown_file.write("{}\n".format(var_tipo_j))
46         print("Campos no Join:")
47         var_campo_j = "{}".format(item.attrib["joinOrder"])
48         markdown_file.write("{}\n".format(var_campo_j))
49
50         for atb in item.findall("viewAttributes/
viewAttribute"):
51             print(atb.attrib["id"])
52             var_atb_id = atb.attrib["id"]
53             markdown_file.write("{}\n".format(var_atb_id))
54
55     print("Nos inseridos no Join")
56     markdown_file.write("Nos inseridos no Join")
```

```
57
58     for ipt in item.findall("input"):
59         print(ipt.attrib["node"])
60         var_inp_id = ipt.attrib["node"]
61         markdown_file.write("{}\n".format(var_inp_id))
62
63     print("Conexao do Join")
64     markdown_file.write("Conexao do Join")
65
66     for con in item.findall("joinAttribute"):
67         print(con.attrib["name"])
68         var_con_id = con.attrib["name"]
69         markdown_file.write("{}\n".format(var_con_id))
70
71     else:
72         print("Node Output")
73         markdown_file.write("Node Output")
74
75     for atb, item in enumerate(root.findall("
76         calculationViews/calculationView/viewAttributes/
77         viewAttribute")):
78         print(" {}: {}".format(atb, item.attrib["id"]))
79         var_node_id = " {}: {}".format(atb, item.attrib[
80             "id"])
81         markdown_file.write("{}\n".format(var_node_id))
82
83     print(" Campos calculados")
84     markdown_file.write(" Campos calculados")
85
86     for calc, item in enumerate(root.findall("
87         calculationViews/calculationView/
88         calculatedViewAttributes/calculatedViewAttribute"
89     )):
90         print(" {}: {} {}".format(calc, item.attrib["id"]
91             ], item.attrib["datatype"]))
```

```
85     var_calc = " {}: {} {}".format(calc, item.attrib
86         ["id"], item.attrib["datatype"])
87     markdown_file.write("{}\n".format(var_calc))
88
89     formula = item.find('formula').text
90     print(" Formula do campo: ", formula)
91     var_form = " Formula do campo: ", formula
92     markdown_file.write("{}\n".format(var_form))
93 # Feche o arquivo Markdown
94 markdown_file.close()
```

Listing B.1 – código fonte Script de Varredura de Calculation View

ANEXO C – Arquivo de Configuração para Execução - *config_doc*

```
1  
2 VARIAVEL;VALOR  
3 DEV_RESPONSAVEL; STEPHANY MENDES.  
4 TITULO;TITULO DO DOCUMENTO  
5 TABELA;NOME_DA_TABELA
```

Listing C.1 – Arquivo de Configuração para Execução

ANEXO D – Pipeline de Desenvolvimento e Integração Contínua de Documentação

```
1
2 trigger:
3   branches:
4     include:
5     - master
6 stages:
7 - stage: __default
8   jobs:
9   - job: Job
10  pool:
11    vmImage: ubuntu-latest
12  steps:
13  - task: Bash@3
14    displayName: 'Build extensions'
15    inputs:
16      targetType: inline
17      script: |
18        if [ -f extensions.csproj ]
19        then
20          dotnet build extensions.csproj --output ./bin
21        fi
22  - task: UsePythonVersion@0
23    displayName: 'Use Python 3.9'
24    inputs:
25      versionSpec: '3.9'
26      addToPath: true
```

```
27     architecture: 'x64'
28   - task: Bash@3
29     displayName: 'Install Application Dependencies'
30     inputs:
31       targetType: inline
32       script: |
33         pip3.9 install -r requirements.txt
34   - task: Bash@3
35     displayName: 'Run Script Documentation'
36     inputs:
37       targetType: inline
38       script: |
39         python generate_data_dictionary.py
40   - task: CopyFiles@2
41     inputs:
42       SourceFolder: '$(Build.SourcesDirectory)'
43       Contents: '*.md'
44       TargetFolder: '$(Build.ArtifactStagingDirectory)'
45   - task: PublishBuildArtifacts@1
46     inputs:
47       PathToPublish: '$(Build.ArtifactStagingDirectory)'
48       ArtifactName: 'drop'
49       publishLocation: 'Container'
50   - task: WikiUpdaterTask@2
51     inputs:
52       repo: 'endereco-do-repositorio'
53       filename: 'ReleaseNotes/$(Build.BuildNumber).md'
54       replaceFile: true
55       dataIsFile: true
56       sourceFile: '$(Build.SourcesDirectory)/
57         DOCUMENTACAO_TESTE.md'
58       message: 'Auto-Generated Release Notes'
59       gitname: '$(Build.RequestedFor)'
60       gitemail: '$(Build.RequestedForEmail)'
61       useAgentToken: false
```

```
61     user: 'devdocumentation-ops'  
62     password: '$(PAT)'  
63     localpath: '$(System.DefaultWorkingDirectory)/repo'  
64     injectExtraHeader: false
```

Listing D.1 – *Script* de Configuração de *Pipeline*