

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

**CATÁLOGO DE MELHORES PRÁTICAS DE
GESTÃO DE PESSOAS EM PROJETOS DE
DESENVOLVIMENTO DE SOFTWARE**

DISSERTAÇÃO DE MESTRADO

Marcelo Falkowski Burkard

Santa Maria, RS, Brasil

2024

CATÁLOGO DE MELHORES PRÁTICAS DE GESTÃO DE PESSOAS EM PROJETOS DE DESENVOLVIMENTO DE SOFTWARE

Marcelo Falkowski Burkard

Dissertação apresentada ao Curso de Mestrado Programa de
Pós-Graduação em Ciência da Computação (PPGCC), Área de Concentração
em Computação Aplicada, da Universidade Federal de Santa Maria (UFSM,
RS),
como requisito parcial para obtenção do grau de
Mestre em Ciência da Computação

Orientadora: Prof^ª. Dr. Lisandra Manzoni Fontoura

Santa Maria, RS, Brasil

2024

Falkowski Burkard, Marcelo

Catálogo de melhores práticas de gestão de pessoas em projetos de desenvolvimento de software / por Marcelo Falkowski Burkard. – 2024.
118 f.: il.; 30 cm.

Orientadora: Lisandra Manzoni Fontoura
Dissertação (Mestrado) - Universidade Federal de Santa Maria,
Centro de Tecnologia, Programa de Pós-Graduação em Ciência da
Computação, RS, 2024.

1. Gestão de pessoas. 2. Problemas. 3. Práticas. 4. Revisão sistemática da literatura. I. Manzoni Fontoura, Lisandra. II. Título.

© 2024

Todos os direitos autorais reservados a Marcelo Falkowski Burkard. A reprodução de partes ou do todo deste trabalho só poderá ser feita mediante a citação da fonte.

E-mail: mburkard@inf.ufsm.br

**Universidade Federal de Santa Maria
Centro de Tecnologia
Programa de Pós-Graduação em Ciência da Computação**

A Comissão Examinadora, abaixo assinada,
aprova a Dissertação de Mestrado

**CATÁLOGO DE MELHORES PRÁTICAS DE GESTÃO DE PESSOAS
EM PROJETOS DE DESENVOLVIMENTO DE SOFTWARE**

elaborada por
Marcelo Falkowski Burkard

como requisito parcial para obtenção do grau de
Mestre em Ciência da Computação

COMISSÃO EXAMINADORA:

Lisandra Manzoni Fontoura, Dr.
(Presidente/Orientadora)

Gilnei Luiz De Moura, Dr. (UFSM)

Cristiano Tolfo, Dr. (UFSC)

Santa Maria, 09 de Janeiro de 2024.

À minha filha ou meu filho, Yasmin ou Benício, que começou a chutar na barriga da mãe enquanto concluo este trabalho.

AGRADECIMENTOS

Obrigado aos meus pais Nestor (in memoriam) e Wanda por todas as oportunidades e sacrifícios que pavimentaram a minha chegada até aqui. Ao meu irmão Janô, por ter plantado as primeiras sementes no gosto pela computação. À minha irmã Juliana pelas incontáveis horas de revisão. À minha esposa Lara por toda a paciência comigo ao longo dos anos. Amo vocês!

"No one told you when to run, you missed the starting gun"
— TIME, PINK FLOYD

RESUMO

Dissertação de Mestrado
Programa de Pós-Graduação em Ciência da Computação
Universidade Federal de Santa Maria

CATÁLOGO DE MELHORES PRÁTICAS DE GESTÃO DE PESSOAS EM PROJETOS DE DESENVOLVIMENTO DE SOFTWARE

AUTOR: MARCELO FALKOWSKI BURKARD

ORIENTADORA: LISANDRA MANZONI FONTOURA

Local da Defesa e Data: Santa Maria, 09 de Janeiro de 2024.

Este estudo investiga práticas de gestão de pessoas em projetos de desenvolvimento de software, com o objetivo de elaborar um catálogo de melhores práticas para uso por pesquisadores e profissionais da área. A pesquisa destaca a importância crítica das pessoas na execução desses projetos, tanto sob abordagens tradicionais quanto ágeis. O trabalho realiza uma revisão sistemática da literatura para identificar problemas comuns na gestão de pessoas em projetos de software e as práticas eficazes para solucioná-los. Foram utilizados quatro engines de busca: ACM, IEEE, Scielo e ScienceDirect filtrando estudos primários em língua inglesa publicados entre 2016 e 2023. Partindo de 2495 estudos primários não duplicados, foi realizada uma análise detalhada de 63 estudos, que resultou na identificação de nove agrupamentos de problemas e dezesseis práticas distintas. Essas práticas foram validadas por 31 especialistas, permitindo classificá-las em termos de relevância. Os resultados revelam a preeminência das habilidades interpessoais (soft skills) sobre as técnicas (hard skills) e enfatizam a importância de práticas como feedback contínuo, comunicação aberta e gestão transparente. Este catálogo serve como um recurso para gestores avaliarem e adaptarem práticas à realidade específica de seus projetos.

Palavras-chave: Gestão de pessoas. Problemas. Práticas. Revisão sistemática da literatura.

ABSTRACT

Master's Dissertation
Post-Graduate Program in Informatics
Federal University of Santa Maria

CATALOG OF BEST PEOPLE MANAGEMENT PRACTICES IN SOFTWARE DEVELOPMENT PROJECTS

AUTHOR: MARCELO FALKOWSKI BURKARD

ADVISOR: LISANDRA MANZONI FONTOURA

Defense Place and Date: Santa Maria, January 09th, 2024.

This study investigates people management practices in software development projects, aiming to compile a catalog of best practices for researchers and professionals in the field. The research highlights the critical importance of people in the execution of these projects, under both traditional and agile approaches. The work conducts a systematic literature review to identify common problems in people management in software projects and the effective practices to solve them. Utilizing four search engines: ACM, IEEE, Scielo and ScienceDirect, filtering primary studies in english language published between 2016 and 2023. Starting with 2495 non-duplicated primary studies, a detailed analysis of 63 studies was conducted, resulting in the identification of nine clusters of problems and sixteen distinct practices. These practices were validated by 31 experts, allowing them to be classified in terms of relevance. The findings reveal the predominance of interpersonal skills (soft skills) over technical skills (hard skills) and emphasize the importance of practices such as continuous feedback, open communication, and transparent management. This catalog serves as a resource for managers to assess and adapt practices to the specific reality of their projects.

Keywords: People Management, Problems, Practices, Systematic Literature Review.

LISTA DE TABELAS

Tabela 3.1 – Termos de busca e equivalências em inglês.....	20
Tabela 3.2 – String de busca utilizada igualmente nos engenhos selecionados.	20
Tabela 3.3 – Critérios de inclusão e exclusão.	21
Tabela 3.4 – Quantidade de resultados por engenho.	21
Tabela 4.1 – Quantidade de artigos por natureza dos problemas.....	26
Tabela 4.2 – Quantidade de artigos por grupo de práticas.....	33
Tabela 5.1 – Informações demográficas dos respondentes do questionário	46
Tabela 5.2 – Respostas dos participantes sobre relevância a das práticas	50
Tabela 5.3 – Ranking de práticas ordenadas por relevância.....	84
Tabela 5.4 – Relevância geral por prática e relevância para cada problema associado	85
Tabela 5.5 – Práticas com maior relevância para cada tipo de problema-alvo	86
Tabela 5.6 – Quantidade de problemas alvo	86
Tabela 5.7 – Problemas-alvo correlacionados com as 3 práticas mais relevantes	87
Tabela 6.1 – Modelo utilizado para descrição das práticas.	89
Tabela 6.2 – Prática 001 - Usar práticas ágeis.....	90
Tabela 6.3 – Prática 002 - Avaliar traços de personalidade e soft skills.	91
Tabela 6.4 – Prática 003 - Executar checklist de onboarding.	92
Tabela 6.5 – Prática 004 - Criar canais e promover a comunicação aberta.	93
Tabela 6.6 – Prática 005 - Criar cronograma mínimo.....	94
Tabela 6.7 – Prática 006 - Criar job description para contratação.....	95
Tabela 6.8 – Prática 007 - Prover feedback ao time.....	96
Tabela 6.9 – Prática 008 - Definir tamanho do time.	97
Tabela 6.10 – Prática 009 - Desafiar time a aprender novas habilidades.....	98
Tabela 6.11 – Prática 010 - Praticar gestão à vista.	99
Tabela 6.12 – Prática 011 - Identificar e gerenciar competências do time.	100
Tabela 6.13 – Prática 012 - Montar plano de carreira e plano de sucessão.....	101
Tabela 6.14 – Prática 013 - Montar time heterogêneo.....	102
Tabela 6.15 – Prática 014 - Organizar treinamentos.	103
Tabela 6.16 – Prática 015 - Rotacionar time.	104
Tabela 6.17 – Prática 016 - Usar indicadores para monitorar a performance do time de desenvolvimento de software.	105

SUMÁRIO

1 INTRODUÇÃO	13
1.1 Justificativa e motivação	13
1.2 Objetivos	14
1.3 Estrutura do texto	14
2 REFERENCIAL TEÓRICO	15
2.1 Gestão de projetos de desenvolvimento de software	15
2.2 Gestão de pessoas em projetos de desenvolvimento de software	16
2.3 Melhores práticas	17
3 METODOLOGIA	19
3.1 Protocolo de revisão	19
3.1.1 Questões de pesquisa	19
3.1.2 String de busca	20
3.1.3 Regras de inclusão e exclusão	20
3.1.4 Processo de busca primária	21
3.1.5 Processo de seleção dos estudos	23
3.2 Extração de resultados e catálogo de práticas	24
3.3 Validação com especialistas	24
4 REVISÃO SISTEMÁTICA DA LITERATURA	26
4.1 Questão de pesquisa 1	26
4.1.1 Comunicação	26
4.1.2 Motivação	27
4.1.3 Habilidades e conhecimento técnico	28
4.1.4 Aspectos geográficos	29
4.1.5 Comprometimento	29
4.1.6 Satisfação	30
4.1.7 Estabilidade do time	30
4.1.8 Foco	31
4.1.9 Autonomia	32
4.2 Questão de pesquisa 2	32
4.2.1 Montagem de time	33
4.2.2 Treinamento	36
4.2.3 Comunicação	38
4.2.4 Onboarding	39
4.2.5 Emprego de práticas ágeis	40
4.2.6 Otimização de alocação	41
4.2.7 Job rotation	42
4.2.8 Monitoramento de performance	43
5 VALIDAÇÃO	45
5.1 Dados demográficos	45
5.2 Práticas	49
5.2.1 Prática 001 - Usar práticas ágeis	51
5.2.2 Prática 002 - Avaliar traços de personalidade e soft skills	53
5.2.3 Prática 003 - Executar checklist de onboarding	55
5.2.4 Prática 004 - Criar canais e promover a comunicação aberta	58
5.2.5 Prática 005 - Criar cronograma mínimo	60

5.2.6 Prática 006 - Criar job description para contratação	61
5.2.7 Prática 007 - Prover feedback ao time	63
5.2.8 Prática 008 - Definir tamanho do time	65
5.2.9 Prática 009 - Desafiar time a aprender novas habilidades	67
5.2.10 Prática 010 - Praticar gestão à vista	69
5.2.11 Prática 011 - Identificar e gerenciar competências do time	72
5.2.12 Prática 012 - Montar plano de carreira e plano de sucessão	74
5.2.13 Prática 013 - Montar time heterogêneo	75
5.2.14 Prática 014 - Organizar treinamentos	78
5.2.15 Prática 015 - Rotacionar time	80
5.2.16 Prática 016 - Usar indicadores para monitorar a performance do time de desenvolvimento de software	82
5.3 Discussão	84
6 CATÁLOGO DE PRÁTICAS	89
7 CONCLUSÃO	106
7.1 Considerações finais	106
7.2 Limitações do estudo	107
7.3 Trabalhos futuros	108
REFERÊNCIAS	109

1 INTRODUÇÃO

1.1 Justificativa e motivação

Pessoas são um dos elementos fundamentais para a execução de projetos de desenvolvimento de software, a sua importância é reconhecida tanto pelas abordagens tradicionais quanto pelas abordagens ágeis para gestão de projetos.

Na abordagem tradicional, o PMBOK (PMI 2017) possui uma área inteira dedicada à gestão de pessoas com atividades específicas para o desenvolvimento da equipe, melhoria de competências individuais, melhoria do ambiente geral e melhoria das interações entre o time; objetivando o aumento da motivação dos profissionais, o trabalho em equipe, a diminuição de *turnover* e, por fim, a melhoria nos resultados alcançados.

Convergente a isso, no Manifesto Ágil (Beck et al. 2001) os autores afirmam que, embora haja valor no uso de processos e ferramentas, devemos valorizar e priorizar os indivíduos e como eles interagem, pois a performance de times de desenvolvimento depende da forma com que as pessoas se relacionam.

É possível observar na literatura relacionada à engenharia de software grande volume de trabalhos relacionados a tecnologias e processos. Comparativamente pouco foi feito em relação à gestão de pessoas, havendo poucos estudos empíricos explorando esse tipo de gestão.

De acordo com Mishra e Misra (2010) o sucesso dos projetos depende, em grande parte, da iniciativa dos indivíduos que compõem o time, sua mentalidade, aplicação do conhecimento e abordagem sistemática. Nessa perspectiva, as organizações devem buscar e empregar as melhores práticas na gestão de pessoas a fim de alcançar a excelência nos resultados produzidos.

Melhores práticas levam as organizações a terem um desempenho cada vez mais alto. De acordo com Kerzner (2006) melhores práticas levam a melhorias contínuas, que, por sua vez, levam à adoção de novas melhores práticas. A partir disso, pode-se inferir que o que determina se uma prática é ou não a melhor varia de acordo com o contexto onde ela será aplicada. Cabe, portanto, ao gestor avaliar quando e onde aplicá-la.

Este trabalho busca criar um novo artefato para uso dos gestores de projetos de software na prática de suas atividades compilando a literatura recente e facilitando a busca por práticas que se adéquem ao contexto de trabalho de cada gestor.

1.2 Objetivos

O presente trabalho tem como objetivo geral elaborar um catálogo de melhores práticas para auxiliar gestores de projeto de desenvolvimento de software em suas atividades de gestão de pessoas. Os objetivos específicos incluem:

- Explorar a gestão de pessoas em projetos de desenvolvimento de software
- Investigar as características que definem "melhores práticas"
- Mapear práticas de gestão de pessoas em projetos de desenvolvimento de software
- Validar melhores práticas com especialistas da área
- Elaborar documento detalhando cada uma das melhores práticas

1.3 Estrutura do texto

Este trabalho inicia apresentado seu objetivo e relevância para o campo estudo. O Capítulo 2 aborda a base teórica do estudo. A metodologia, incluindo procedimentos para seleção e análise dos dados, além da validação por especialistas, é detalhada no Capítulo 3. Os resultados da revisão sistemática da literatura, respondendo às questões de pesquisa, são expostos no Capítulo 4. O processo de validação das práticas identificadas é examinado no Capítulo 5, onde são discutidos os resultados e opinião dos especialistas. O Capítulo 6 descreve o formato para a descrição das práticas e as apresenta. Concluindo, o Capítulo 7 reflete sobre os resultados, aborda as limitações do estudo e propõe direções para futuras pesquisas, com o intuito de ampliar o entendimento e aplicação das práticas de gestão de pessoas em projetos de software.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta os conceitos fundamentais para o entendimento do trabalho, apresentando a visão de diferentes autores acerca do tema.

2.1 Gestão de projetos de desenvolvimento de software

O Project Management Body of Knowledge (PMBOK) é uma fonte de orientação importante para profissionais e organizações envolvidas no gerenciamento de projetos, especialmente no desenvolvimento de software. Este conjunto de práticas reconhecidas internacionalmente oferece diretrizes, processos e melhores práticas que ajudam a estruturar e gerenciar projetos de software de maneira eficiente.

Ao adotar os princípios e processos delineados no PMBOK, os gerentes de projetos de software podem melhorar a entrega dentro do prazo, do orçamento e da qualidade, além de mitigar riscos e maximizar o valor agregado ao cliente. Em um cenário de evolução tecnológica e demandas crescentes por produtos de software de alta qualidade, a compreensão e aplicação do PMBOK se tornam vitais para garantir o sucesso dos projetos de desenvolvimento de software.

A sétima edição do PMBOK (PMI 2021) apresenta alguns conceitos chave:

- **Projeto.** Um empreendimento temporário empreendido para criar um produto, serviço ou resultado único. A natureza temporária dos projetos indica um início e um fim para o trabalho do projeto ou uma fase do trabalho do projeto. Os projetos podem ser independentes ou fazer parte de um programa ou portfólio.
- **Gestão de projetos.** A aplicação de conhecimentos, habilidades, ferramentas e técnicas às atividades do projeto para atender aos requisitos do projeto. A gestão de projetos refere-se a orientar o trabalho do projeto para entregar os resultados pretendidos. As equipes de projeto podem alcançar os resultados usando uma ampla variedade de abordagens (por exemplo, preditiva, híbrida e adaptativa).
- **Gerente de projeto.** A pessoa designada pela organização executora para liderar a equipe do projeto responsável por alcançar os objetivos do projeto. Os gerentes de projeto desempenham uma variedade de funções, como facilitar o trabalho da equipe do projeto para alcançar os resultados e gerenciar os processos para entregar os resultados pretendidos.

- Equipe do projeto. Um conjunto de indivíduos que realizam o trabalho do projeto para alcançar seus objetivos.

Ao reconhecer a importância da equipe de projeto, é essencial que o gerente de projeto aborde estratégias de gestão de pessoas que maximizem as chances de sucesso do empreendimento.

2.2 Gestão de pessoas em projetos de desenvolvimento de software

O Processo de desenvolvimento de software é o conjunto de atividades que leva à produção de um produto de software. As atividades desse processo, como toda atividade criativa e intelectual, são complexas e dependem do julgamento humano (Sommerville 2010). De acordo com Alcântara (2018), sem uma boa gestão de pessoas o resultado do projeto é, em geral, inadequado.

Desde 1994, o Software Engineering Institute (SEI) publica modelo de maturidade de gestão de pessoal P-CMM - *People Capability Maturity Model* (Curtis, Hefley e Miller 2009), com objetivo de ajudar organizações na gestão de pessoas em projetos de desenvolvimento de software: atrair, motivar, entregar e reter talentos necessários para o desenvolvimento e entrega de software. O modelo possui cinco níveis de maturidade com práticas em múltiplas áreas de prática: recrutamento, seleção, gestão de performance, treinamento, compensações, desenvolvimento de carreira, desenho organizacional e desenvolvimento de cultura de time. Johnson (2008) complementa que gestores devem ser líderes habilidosos e bons organizadores de sessões para resolução de problemas maximizando colaboração entre departamentos.

Para que o time possa entregar o melhor trabalho, Mishra e Misra (2010) listam como necessário: satisfação no trabalho, moral elevado, ambiente de trabalho saudável, forte apoio monetário, tarefas orientadas à sua carreira, educação e treinamento adequado.

Lalsing, Kishnah e Pudaruth (2012) identifica alguns fatores humanos que influenciam na produtividade e qualidade de projetos de desenvolvimento de software: comunicação, colaboração, liderança, empoderamento, confiança e motivação. Demir (2017) menciona a importância da formação adequado do time e trabalho em equipe com envolvimento e comprometimento para alcançar os objetivos esperados.

Analisando fatores humanos que influenciam a entrega de projetos de desenvolvimento de software, Caulo et al. (2021a) apresenta relações entre traços de personalidade e produtivi-

dade dos indivíduos em times de desenvolvimento de software, o gestor pode, portanto, empregar técnicas como o teste DISC¹ para mapear o perfil dos indivíduos e compor uma combinação de membros mais adequada para alcançar sucesso nos projetos (Xia, David Lo et al. 2017).

São diversas as técnicas e práticas que podem ser aplicadas no dia-a-dia de times de desenvolvimento de software, cabe ao gestor garantir que as melhores sejam aplicadas para o sucesso dos projetos.

2.3 Melhores práticas

Quanto ao significado de melhores práticas, em Tighy (2012, p. 1) encontram-se duas possíveis definições:

- “Uma melhor prática é um método bem definido que contribui para uma etapa bem sucedida no desenvolvimento de um produto”;
- “Um método ou técnica que consistentemente demonstra resultados superiores aos alcançados através de outros meios”.

Referente a melhores práticas na área de gestão de projetos de desenvolvimento de software Kerzner (2006) afirma que não há consenso entre as empresas quanto a quais práticas são as melhores.

A metodologia para identificação de melhores práticas pode partir de análise e comparação de resultados disponíveis na literatura, em von Wangenheim et al. (2010) pesquisadores com expertise prática e teórica buscam as melhores práticas nos trabalhos já publicados sobre o assunto existente, compilam resultados e, posteriormente submetem para revisão de outros pesquisadores ao longo de diversas rodadas até alcançarem consenso.

Ng (2018) cita o PMBOK (PMI 2017) como um exemplo, um guia com 47 processos ou melhores práticas, dentre os quais, cabe ao gestor praticante selecionar e adaptar (do inglês *tailoring*) aqueles modelos que melhor se aplicam à sua realidade.

Segundo Warrick (2017), a cultura organizacional pode influenciar significativamente o desempenho e a eficácia de uma empresa; o moral e a produtividade de seus funcionários. Portanto, pode-se afirmar que o gerente de projeto deve observar aspectos da cultura organizacional

¹ Acrônimo em inglês das quatro perfis de personalidade caracterizados no teste: Dominância, Influência, Estabilidade e Cautela

e especificidades de cada organização ao selecionar ou adaptar as melhores práticas de gestão de pessoas.

3 METODOLOGIA

A metodologia do presente trabalho envolve o emprego inicial de revisão sistemática literária (RSL), pois ajuda a obter um panorama geral sobre os estudos existentes em um domínio específico. Esta é dividida em duas etapas:

Na primeira etapa foi realizado planejamento incluindo proposta e definição do protocolo a ser seguido baseado no proposto por Kitchenham e Charters (2007): fontes de dados, strings de busca, critérios para escolha do material utilizado na revisão e processo de busca e extração de dados.

Durante a segunda etapa, foi realizada a busca por artigos científicos publicados entre 2016 e 2023, selecionados de acordo com o protocolo estabelecido, extração de dados e metadados dos artigos e realizada síntese do material apresentado no Capítulo 4.

Após a conclusão da revisão sistemática, foi gerado um catálogo de práticas que foi validado através de questionário com especialistas em gestão de projetos de desenvolvimento de software. Cada prática foi descrita em formato baseado no utilizado por Telemaco et al. (2020) apud Beck (1995).

A imagem 3.1 ilustra o processo para realização do trabalho.

3.1 Protocolo de revisão

O protocolo de revisão está pautado em questões de pesquisa, string de busca, regras de inclusão e exclusão, processo de busca primária e processo de seleção dos estudos, os quais seguem detalhados.

3.1.1 Questões de pesquisa

O objetivo deste trabalho é realizar uma revisão sistemática de literatura mapeando práticas de gestão de pessoas em projetos de desenvolvimento de software.

Como metas específicas, buscou-se na literatura recente, contemplando as publicações entre 2016 e 2023, respostas para as seguintes questões:

- RQ1: Que problemas relacionados à gestão de pessoas que afetam os projetos de desenvolvimento de software foram citados pelo autor do estudo primário?

- RQ2: Quais práticas relacionadas à gestão de pessoas foram citadas pelo autor do estudo primário e que tendem a solucionar problemas de projetos de desenvolvimento de software?

3.1.2 String de busca

A partir destas palavras-chave foram identificados termos de busca presentes na Tabela 3.1.

Palavras-chave	Termos de busca
Desenvolvimento de software	Software development, software project, software engineering, information system development
Fatores Humanos	Human resource, human resources, people factor, resource management, team
Produtividade	Productivity, performance, velocity

Tabela 3.1 – Termos de busca e equivalências em inglês.

Foram conduzidos três ciclos de busca *ad hoc* no engenho IEEE Xplore. Neles avaliaram-se os artigos mais relevantes sugeridos pelo engenho e foram descartadas as palavras-chave não encontradas em nenhum desses artigos. Ao final foi obtida uma lista reduzida de palavras-chave para criação da *string* de busca.

A busca nos engenhos ocorreu por meio de uma *string*, para a qual foi gerada uma *string* única que pode ser utilizada igualmente nos quatro engenhos selecionados:

String de busca

("Software Development"OR "Software Project"OR "Software Engineering") AND ("Human Resource"OR "Human Resources"OR "People Factor") AND ("Productivity"OR "Performance")

Tabela 3.2 – String de busca utilizada igualmente nos engenhos selecionados.

3.1.3 Regras de inclusão e exclusão

Para identificar fontes relevantes ao presente trabalho foram definidos critérios para selecionar quais estudos primários são incluídos e quais são excluídos deste. Os critérios de inclusão e exclusão utilizados nessa pesquisa foram baseados no proposto por Kitchenham e Charters (2007) com ajustes de acordo com as questões de pesquisa dessa RSL. A Tabela 3.3 apresenta os critérios adotados:

ID	Critério de inclusão	ID	Critério de exclusão
CI1	Estudo primário	CE1	Estudo não é estudo primário
CI2	Estudo relacionado a desenvolvimento de software	CE2	Estudo não responde às questões de pesquisa
CI3	Estudo relacionado a fatores humanos	CE3	Estudo que se apresenta como livro
CI4	Estudo publicado entre 2016 e 2023	CE4	Estudo publicado antes de 2016
CI5	Estudo publicado em língua inglesa	CE5	Estudo não publicado em língua inglesa
		CE6	Estudo duplicado
		CE7	Estudo com menos de 15 pontos de score

Tabela 3.3 – Critérios de inclusão e exclusão.

3.1.4 Processo de busca primária

A busca primária foi executada em três bases de dados: Association for Computing Machinery - ACM Digital Library ², IEEE Xplore ³, Scielo ⁴ e ScienceDirect ⁵, bibliotecas digitais online que são vastamente utilizadas por pesquisadores da áreas de ciências exatas e humanas. Foram consideradas apenas bibliotecas que pudessem exportar arquivos no formato *bibtex* com resumo incluído para simplificar o processo de análise.

A busca foi realizada considerando um período delimitado para os últimos cinco anos completos, ou seja, entre 2016 e 2022, e as publicações de 2023 disponibilizadas até o momento da realização da consulta⁶. Foram encontrados 2736 artigos nos quatro engenhos:

Engenho	Resultados encontrados
ACM	969
IEEE	328
Scielo	2
ScienceDirect	1437

Tabela 3.4 – Quantidade de resultados por engenho.

Os resultados foram exportados em arquivos no formato *bibtex* incluindo resumos e todos metadados para posterior análise.

² <https://dl.acm.org/>

³ <https://ieeexplore.ieee.org/>

⁴ <https://www.scielo.org/>

⁵ <https://www.sciencedirect.com/>

⁶ Consultas realizadas em 06 de julho de 2023

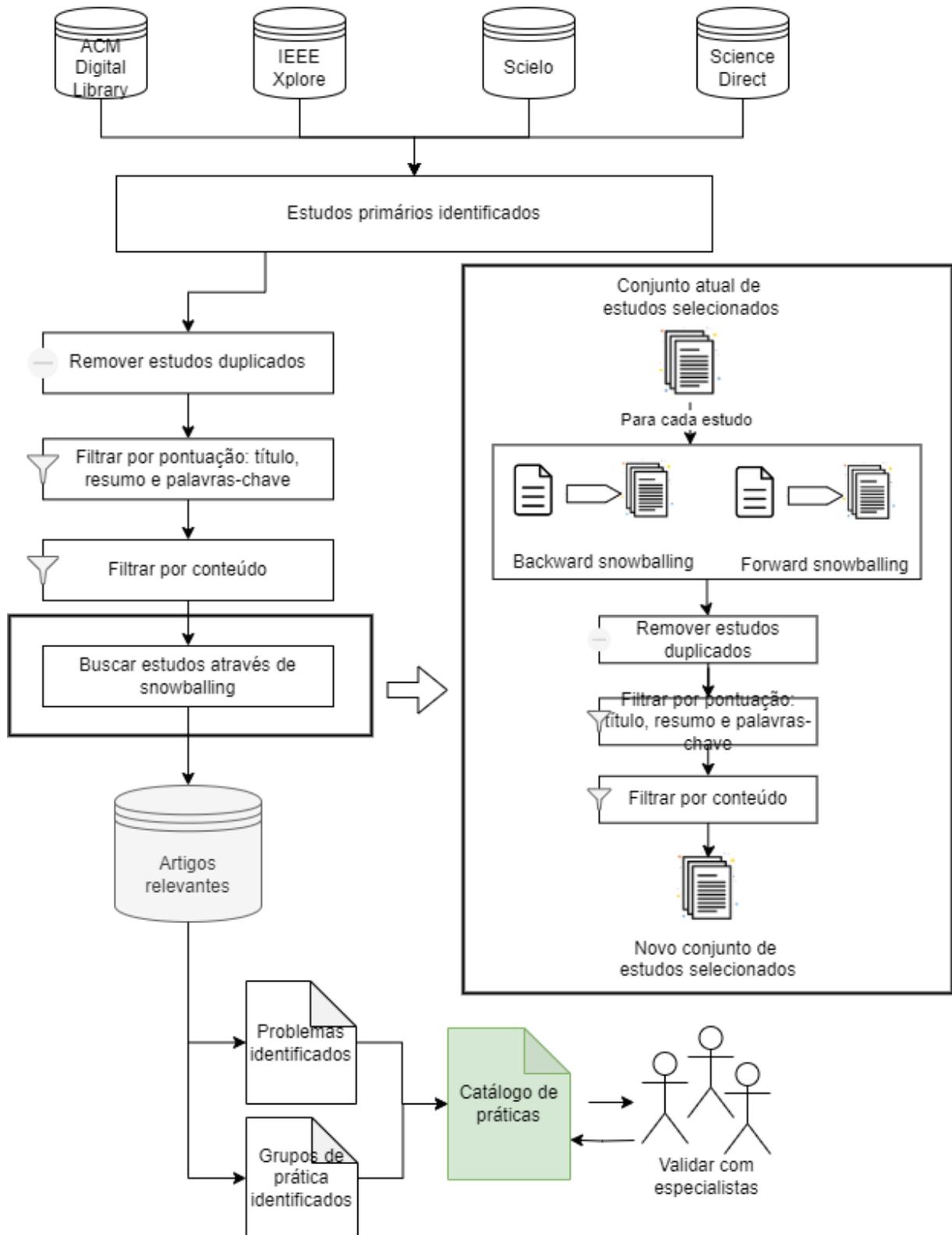


Figura 3.1 – Processo para realização do trabalho

3.1.5 Processo de seleção dos estudos

A seleção dos estudos foi dividida em quatro etapas:

Na primeira etapa os arquivos *bibtex* foram importados na ferramenta StArt⁷, que identificou automaticamente 241 artigos duplicados sendo eles rejeitados de acordo com o critério CE6, restando 2495 artigos não duplicados. Foram também identificados 239 artigos exportados pelo engenho, porém, publicados antes de 2016 ou com data de publicação ausente, sendo eles também rejeitados de acordo com o critério CE4. A ferramenta StArt atribui score para cada um dos artigos de acordo com o número de ocorrências de palavras-chave no título (5 pontos), resumo (3 pontos) e palavras-chave (2 pontos); 2051 artigos com menos de 15 pontos foram rejeitados nesse momento de acordo com o critério CE7.

Na segunda etapa, foram observados título e resumo dos artigos restantes considerando os critérios de inclusão e exclusão. O autor leu individualmente cada um dos artigos classificando-os de acordo com os critérios de inclusão e exclusão, 134 artigos foram rejeitados de acordo com o critério CE2, resultando em 71 artigos selecionados ao final desta etapa.

Na terceira etapa os artigos foram lidos avaliando sua introdução, metodologia e conclusão, sendo excluídos 36 de acordo com os critérios CE1, CE2, CE3 e CE5. Nesse momento 35 estudos foram selecionados. Nesse momento foram respondidas as questões de pesquisa e os resultados parciais do trabalho foram publicados em formato de artigo (Burkard e Fontoura 2023) na conferência ICEIS 2023⁸ e os comentários dos especialistas foram incorporados ao restante do trabalho.

Na quarta etapa foi realizado processo de snowballing, uma técnica usada para identificar estudos adicionais a partir das referências dos artigos identificados, iniciamos a busca pelos artigos que foram referenciados e que também referenciam os 35 artigos iniciais.

Primeiramente, foram identificadas referências a 82 novos artigos potencialmente relevantes para a pesquisa. Esses artigos foram catalogados e obtidos para avaliação. Cada um desses 82 artigos foi avaliado de acordo com o mesmo processo das etapas 1, 2 e 3, considerando sua relevância e adequação aos critérios de inclusão e exclusão previamente estabelecidos para esta pesquisa. Dessa forma, 54 artigos foram excluídos, restando 28 artigos considerados relevantes após o processo de snowballing.

Somando os 28 artigos identificados pelo snowballing aos 35 artigos originalmente sele-

⁷ State of the Art through Systematic Review - disponível em <http://lapes.dc.ufscar.br>

⁸ 25th International Conference on Enterprise Information Systems

cionados, a pesquisa passou a ser composta por um total de 63 artigos. Estes artigos formaram a base da revisão e foram utilizados para leitura integral, extração de dados, análise e composição dos resultados finais da pesquisa.

3.2 Extração de resultados e catálogo de práticas

Os 63 estudos primários selecionados foram lidos de forma integral e organizados de forma a responder às questões de pesquisa. Os problemas e práticas relacionados à gestão de pessoas identificadas foram extraídos, organizadas e categorizadas com base em suas semelhanças.

Na RQ2, dentro de cada categoria identificada, práticas específicas foram formuladas em um formato acionável para uso prático. Para cada prática, foi elaborado um texto descritivo e sumário, consolidando as informações extraídas dos estudos primários, proporcionando uma compreensão abrangente e aplicável das práticas de gestão de pessoas no contexto de projetos de desenvolvimento de software.

Após elaboração das práticas, foram estabelecidas correlações entre cada prática específica e uma ou mais categorias de problemas que a prática visa resolver. Esse material foi submetido a processo para validação de relevância de cada uma das práticas com especialistas na área de desenvolvimento de software.

3.3 Validação com especialistas

Para validar os resultados obtidos na revisão da literatura e o catálogo de práticas propostas, realizamos uma pesquisa com profissionais da área, baseada em entrevistas estruturadas (Kajornboon 2005). O restante da seção apresenta detalhadamente a pesquisa, que seguiu o protocolo proposto por Oishi (2003).

Para a seleção dos participantes, foram estabelecidos critérios específicos. Cada participante deveria ter, no mínimo, um ano de experiência em projetos de desenvolvimento de software, abrangendo funções como gestão, análise, desenvolvimento ou garantia de qualidade. A seleção foi realizada através das redes profissionais e acadêmicas do autor. Para manter a neutralidade dos resultados, pessoas que já tivessem conhecimento prévio sobre esta pesquisa, incluindo coautores e colegas de academia, foram excluídas da lista de participantes.

O questionário enviado aos especialistas possui duas partes, a primeira com perguntas

demográficas e a segunda com perguntas relacionadas à relevância das práticas.

A primeira parte possui quatro perguntas demográficas com objetivo de identificar o perfil dos respondentes, todas de múltipla escolha, com os participantes orientados a selecionar aquela opção que se encaixe melhor à sua realidade:

- Q1: Qual é a sua área de atuação na organização em que trabalha?
- Q2: Qual a sua área de atuação?
- Q3: Qual a sua experiência profissional?
- Q4: Qual o seu maior título escolar?

A segunda parte do questionário possui 16 seções de perguntas, cada seção relacionada a uma prática específica, contendo a descrição da prática, questão sobre a relevância geral da prática e uma questão sobre a relevância da prática para cada um dos problemas alvo correlacionados identificados. Cada pergunta possui quatro opções de resposta:

- (a) Não relevante (0 pontos)
- (b) Pouco relevante (1 ponto)
- (c) Muito relevante (2 pontos)
- (d) Absolutamente relevante (3 pontos)

Cada resposta tem um valor associado que varia de 0 a 3 (com base na relevância da prática na opinião do participante) e que é utilizado para calcular a relevância da prática.

A relevância de uma prática, para um determinado participante, é definida pela resposta da primeira questão cada sessão do formulário. A relevância de uma prática é a soma da relevância para todos os participantes dividido pela quantidade de participantes multiplicado por 3 (pontuação máxima), o resultado será um valor a ser apresentado percentualmente, conforme demonstrado na fórmula abaixo:

$$\text{Relevância} = \frac{\sum_{p=1}^n \text{Relevancia}_p}{n \times 3}$$

Da mesma forma, para cada problema alvo correlacionado à prática é definido pela soma da relevância respondida por cada participante.

4 REVISÃO SISTEMÁTICA DA LITERATURA

Seguindo o protocolo de revisão foram selecionados 63 estudos primários para extração de dados. A presente seção apresenta os resultados encontrados para cada uma das questões de pesquisa.

4.1 Questão de pesquisa 1

RQ1: Que problemas relacionados à gestão de pessoas que afetam os projetos de desenvolvimento de software foram citados pelo autor do estudo primário?

Foram encontrados os mais diversos problemas citados na literatura referenciados pelos autores dos estudos primários, estes, foram analisados e categorizados em *nove* grupos de acordo com as suas semelhanças de forma a sumarizar e simplificar o entendimento do leitor. A tabela 4.1 apresenta a quantidade de ocorrências para cada grupo considerando a natureza dos problemas.

Natureza dos problemas	Quantidade de artigos
Comunicação	7
Motivação	7
Habilidades e conhecimento técnico	7
Aspectos geográficos	5
Comprometimento	4
Satisfação	4
Estabilidade do time	3
Foco	3
Autonomia	2

Tabela 4.1 – Quantidade de artigos por natureza dos problemas.

A seguir cada um dos grupos é abordado de forma individual apresentando as situações relacionados à natureza do problema e suas implicações.

4.1.1 Comunicação

Comunicação refere-se a como um pessoa relaciona-se com outra. Em projetos de desenvolvimento de software, a forma de comunicação facilita o fluxo apropriado de informação entre os membros, melhorando os resultados do projeto. Nesse sentido, comunicação inadequada prejudica os resultados do projeto, e isso pode acontecer por múltiplos fatores.

Shameem et al. (2020) e Margareth e Mulyanto (2021) citam que a escolha incorreta das ferramentas de comunicação pode causar quebras de comunicação e desentendimento entre a mensagem enviada e o entendimento do receptor.

Outro fator que pode gerar comunicação inadequada é o tamanho do time. Quanto maior o número de membros do time, maior é a quantidade de caminhos de comunicação. Magalhães (2017), Wang, Huang e Liu (2018), Bass (2016), Machuca-Villegas et al. (2022) e Shameem et al. (2020) argumentam que um time muito grande tem a comunicação dificultada e, portanto, esta ocorre com frequência insuficiente. E, para que a comunicação ocorra com frequência adequada, Stylianou e Andreou (2016) citam que há aumento de *overhead* de comunicação, isto é, aumento de tempo gasto organizando a comunicação ao invés de realizando trabalho produtivo.

A entrega de resultados é inversamente proporcional ao desperdício gerado por um time, uma forma de minimizá-los é a realização de pontos de checagem frequentes através dos quais pode-se comunicar dentro e fora do time, pontos de acerto e que devem ser replicados ou pontos de melhoria a serem corrigidos. Magalhães (2017) apresenta que a falta de *feedbacks* adequados é um dos principais causadores de desmotivação e esgotamento do time de desenvolvimento.

4.1.2 Motivação

De acordo com a sua definição, motivação move uma pessoa em direção à ação (Machuca-Villegas et al. 2022). As motivações dos indivíduos são incontáveis e podem ser classificadas em motivações intrínsecas e extrínsecas. Franca, Silva and Sharp (2020) argumentam que motivação refere-se ao desejo de trabalhar e é sinalizado pela atitude das pessoas em relação ao seu trabalho, influenciando diretamente a sua performance. A falta de motivação é uma ameaça tanto para o time de desenvolvimento quando para o time de gestão (García et al. 2017).

Sobre as razões para baixa motivação, Magalhães (2017) cita pouca autonomia (Seção 4.1.9) e falta de feedback adequado. Bass et al. (2018) complementa: políticas de emprego, equilíbrio entre vida pessoal e trabalho, baixo desafio técnico, inovação, quantidade de horas, recompensas, bom gerenciamento, condições de trabalho adequadas, envolvimento de trabalho com outros e qualidade do trabalho gerado.

Baixa qualidade do trabalho e dos produtos gerados além de uma das causas, são também apontados por Bass et al. (2018) como um dos efeitos da baixa motivação.

Outros efeitos de baixos índices de motivação geram redução da produtividade, efetivi-

dade e aprendizado do time, de acordo com Fatema e Kazi Sakib (2017). No estudo de Bass et al. (2018) é apresentada correlação entre motivação e turnover nas organizações: profissionais motivados tendem a permanecer em seus empregos atuais enquanto profissionais com baixa motivação tendem a gerar atritos e desligamento.

A motivação dos times depende da escolha de estratégias adequadas. Sobre isso, Sha-meem et al. (2020) comentam que a falta de estratégias motivacionais impacta no trabalho como um todo, inviabilizando a escalada do emprego de metodologias ágeis em desenvolvimento de software.

García et al. (2017) propõe o emprego de *gameificação*, uso de mecânicas e características de jogos para acompanhamento e gestão dos profissionais de desenvolvimento de software como forma de melhorar a sua motivação e engajamento.

4.1.3 Habilidades e conhecimento técnico

O terceiro grupo com maior quantidade de citações na literatura sobre problemas humanos que impactam na entrega de projetos de desenvolvimento de software está atrelado à falta de habilidade ou conhecimento técnico dos membros do time de desenvolvimento, estes geram software com baixa qualidade técnica e consequente aumento do número de bugs e incidentes em um sistema.

Sobre fatores técnicos que influenciam a produtividade em desenvolvimento de software Meyer, Zimmermann e Fritz (2017) listam: domínio de linguagem de programação e ferramentas, tamanho e complexidade do software e qualidade do produto.

Fatema e Kazi Sakib (2017) afirmam selecionar as pessoas com as habilidades técnicas corretas para um projeto como uma das atividades mais complexas. Em contrapartida, Song et al. (2020) e Chiang e Lin (2020) apresentam modelos matemáticos para seleção e alocação otimizada de pessoas levando em consideração as suas habilidades técnicas e que podem tornar essa atividade mais direta e simples.

Nigar (2017) e Cunha et al. (2016) citam que não basta apenas possuir conhecimento técnico, a falta de habilidade (experiência ou prática) na aplicação dos conhecimentos também impacta negativamente na entrega de software.

Em relação a evolução de conhecimento e habilidades, Kula et al. (2021) argumenta que maior estabilidade de time (Seção 4.1.7) está positivamente relacionado com o desenvolvimento de habilidades e conhecimento técnico.

4.1.4 Aspectos geográficos

Especialmente quando se trata de grandes organizações globais ou empresas que fazem uso de *outsourcing* aspectos geográficos passam a desempenhar um importante papel no que diz respeito a produtividade dos times de desenvolvimento. *Outsourcing* é o processo de aquisição de produtos ou serviços de um fornecedor terceirizado. Bass (2016) divide em duas categorias: *Onshore*, em que o fornecedor está localizado no mesmo território que o cliente e *offshore*, que envolve um fornecedor geograficamente remoto, geralmente separado do cliente por uma significativa distância temporal (fuso-horário) e diferença cultural. Em contraste, algumas organizações optam por criar estruturas próprias em formato *offshore*.

O estabelecimento de centros de desenvolvimento *offshore* ajudam as organizações a criar presença em mercados emergentes enquanto beneficiam-se de custos menores, entretanto, Qahtani (2020) e Shameem et al. (2020) alertam que as diferenças culturais impactam negativamente no desempenho dos times.

Relacionado ao sentimento de falta de entrosamento entre os times e dificuldade de colaboração Bass et al. (2018) argumentam que esse sentimento aumenta proporcionalmente com o aumento de distância geográfica e redução de sobreposição dos horários de trabalho causado pelas diferenças de fuso-horário que diminui as janelas de comunicação e gera atrasos na comunicação. Além desses, outro fator citado pelo autor é o desnível de idiomas, o emprego de múltiplas linguagens tem impacto médio/alto no sentimento de distanciamento de times.

A dificuldade de comunicação por diferenças culturais gera um efeito cascata que impacta em outros aspectos descritos no presente trabalho, Britto et al. (2019) exemplificam que baixa adequação cultural pode inclusive dificultar a realização de treinamentos (Seção 4.2.2).

4.1.5 Comprometimento

Comprometimento está relacionado com o engajamento e perseverança. Machuca-Villegas et al. (2022) caracterizam como o nível de responsabilidade individual que uma pessoa assume para realizar suas atividades que compõem a entrega de um time. Kula et al. (2021) identificam que comprometimento é um dos principais fatores que promovem a efetividade do time, e, portanto, ajuda para que as entregas ocorram dentro do prazo. O estudo de Tam et al. (2020) complementa que profissionais com maior nível de comprometimento e expertise técnica tendem a identificar e avaliar melhor os riscos, o que aumenta as chances de sucesso no projeto.

No que tange à produtividade, Oliveira et al. (2016) identificaram que os gestores tendem a perceber maior produtividade nos profissionais que combinam foco, comprometimento e proatividade com tarefas entregues sem atraso e com boa qualidade.

O time deve ser responsável coletivamente pelos objetivos do projeto. Machuca-Villegas et al. (2022) argumentam que todos devem assumir responsabilidade pelos resultados obtidos, cumprir seus deveres e, eventualmente, admitir os próprios erros como forma de melhorar o processo de desenvolvimento de software.

Comprometimento não se limita ao time de desenvolvimento, devendo ser estendido, inclusive, ao cliente. Tam et al. (2020) apresentam que em projetos que seguem metodologia cascata o envolvimento do cliente ocorre principalmente no início e fim do projeto, enquanto em projetos que utilizam metodologias ágeis o grau de envolvimento do cliente deve ser alto durante todo o ciclo de vida do projeto: o cliente também deve ser "ágil".

4.1.6 Satisfação

De acordo com Machuca-Villegas et al. (2022), satisfação no trabalho pode ser determinada pela diferença entre o que a pessoa quer e o que ela tem no seu trabalho, logo, sempre que as expectativas quanto a certos aspectos não são atendidas temos profissionais insatisfeitos. Magalhães (2017) e Chiang e Lin (2020) citam como aspectos que geram satisfação no trabalho: crescimento pessoal e profissional, reconhecimento, oportunidades, salário e relações com colegas e superiores.

Franca, Silva and Sharp (2020) argumentam que satisfação refere-se a emoções prazerosas em relação ao trabalho e influencia diretamente o desempenho de times de desenvolvimento de software pois aumenta o seu desejo em permanecer na organização e presença no trabalho.

4.1.7 Estabilidade do time

Analisando fatores percebidos como contribuintes para entregas dentro do prazo Kula et al. (2021) listam: estabilidade do time, representado por baixo turnover e familiaridade do time, ou seja, a experiência que os membros do time têm trabalhando juntos, a qual pode ser medida pela quantidade de tempo que os membros trabalham uns com os outros. O autor argumenta que ambos fatores estão relacionados com a melhoria de coordenação e capacidade de adaptação em caso de mudanças ambientais, e, que para alcançar melhores entregas os gestores de projetos devem não apenas focar em manter os times estáveis mas também apoiar o time no

desenvolvimento de familiaridade de longa duração.

Um desafio de manter times estáveis por longos períodos de tempo é a rotatividade de funcionários. Visto que a troca de empregos é um fenômeno natural do mercado e alto *turnover* de pessoal tem impacto negativo na produtividade e qualidade durante o desenvolvimento de software, cabe ao gestor mitigar os riscos de turnover. Bass et al. (2018) apresentam correlação entre baixa motivação e altas taxas de *turnover*.

Outro fator contribuinte para taxas de turnover apresentado por Bass et al. (2018) é a natureza do trabalho executado: companhias do setor de *outsourcing* apresentam uma expectativa de *turnover* do time de desenvolvimento maior do que empresas de outros segmentos.

4.1.8 Foco

Um dos perfis de desenvolvedor caracterizados no trabalho de Meyer, Zimmermann e Fritz (2017) é chamado de desenvolvedor focado, este sente-se mais produtivo quando trabalha concentrado em uma única tarefa por vez; e improdutivo quando passam muito tempo em uma tarefa por estarem bloqueados ou em velocidade reduzida. O autor traz algumas sugestões:

- Em nível de time, pode ser benéfico criar espaços físicos mais quietos e menos propenso a interrupções;
- Recomendar que profissionais com desenvolvedor focado a alterar horários de trabalhos para ter maior tempo de trabalho ininterrupto;
- Identificar as preferências de comunicação individuais e evitar reuniões *ad-hoc*.

Reuniões de última hora caracterizam-se como um causador de mudança de foco constante, o que, de acordo com o identificado por Oliveira et al. (2016), impacta no atingimento de estado de "flow" desejado pelos desenvolvedores.

Quando se fala de produtividade, Kohl, Vasilescu e Prikladnicki (2020) entrevistaram desenvolvedores e concluiu que pequenas mudanças de contexto, com duração inferior a 3 minutos não chegam a desconectar o desenvolvedor da tarefa anterior. Por exemplo: executar um script que demora 15 segundos não caracteriza *multitasking* e não gera impactos relacionados a perda de foco.

4.1.9 Autonomia

Autonomia significa capacidade de governar-se pelos próprios meios, de acordo com Fuentes, Frankenberg-Garcia e Newstead (2015). Trazendo para o contexto de projetos de desenvolvimento de software, Machuca-Villegas et al. (2022) elencam exemplos de situações onde o time tem percepção de autonomia:

- Organizar-se para estabelecer e atingir seus objetivos e confiar em suas habilidades para executar as tarefas que são responsáveis;
- Tomar decisões referentes a métodos, técnicas e estratégias para executar suas tarefas;
- Ter liberdade para decidir no que trabalhar, o que investigar e quais ideias seguir;
- Tomar a iniciativa de realizar ações corretivas.

Especialmente no que tange à autonomia do time em definir e atribuir responsabilidades, Magalhães (2017) cita este como fator de alto impacto na performance dos indivíduos durante o desenvolvimento de software: quanto maior a autonomia do time em se auto-gerenciar e definir quem será o responsável por alguma tarefa, maior será a motivação e produtividade do time. Em contraponto a isso, diversos autores argumentam que a atribuição de tarefas é um problema de alocação de recursos, e, portanto, deve ser buscada uma solução otimizada através de modelos matemáticos (vide Seção 4.2.6).

4.2 Questão de pesquisa 2

RQ2: Quais práticas relacionadas à gestão de pessoas foram citadas pelo autor do estudo primário e que tendem a solucionar problemas de projetos de desenvolvimento de software?

Referente às práticas de gestão de pessoas em projetos de desenvolvimento de software, os estudos primários foram analisados e, após extração, as práticas foram agrupadas em *oito* grupos de acordo com suas semelhanças. A Tabela 4.2 apresenta a quantidade de ocorrências de práticas citadas para cada um dos agrupamentos.

Agrupamento de práticas	Quantidade de artigos
Montagem de time	18
Treinamento	16
Comunicação	14
Onboarding	11
Emprego de práticas ágeis	10
Otimização de alocação	9
Job rotation	7
Monitoramento de performance	6

Tabela 4.2 – Quantidade de artigos por grupo de práticas.

A seguir cada um dos grupos é abordado de forma individual apresentando as práticas citadas pelos autores.

4.2.1 Montagem de time

A etapa de formação do time, no qual geralmente o gestor define quem serão os indivíduos que irão compor o time de desenvolvimento é crítica para o sucesso de desenvolvimento de software.

De acordo com Zaouga, Rabai e Alalyani (2019) e Meyer, Zimmermann e Fritz (2017) o gerente deve garantir a correta atribuição de membros a um projeto específico considerando as especificidades técnicas do projeto e capacidade dos indivíduos.

Profissionais subqualificados devem ser evitados, da mesma maneira o gestor deve ter cuidado quanto ao emprego de profissionais superqualificados. Angelis (2019) argumenta que para correta montagem do time deve-se buscar profissionais cuja qualificação seja correspondente à necessidade do projeto. Esse equilíbrio faz com que o projeto flua e que os profissionais tenham um nível adequado de desafio evitando problemas de satisfação e motivação.

Além da importância das habilidades técnicas, Angelis (2019) Caulo et al. (2021b) e Cunha et al. (2016) argumentam que os traços de personalidade dos indivíduos também impactam na performance dos times de desenvolvimento de software. Essas práticas estão descritas com maior profundidade na Seção 4.2.1.

A identificação dos profissionais mais adequados ao projeto inicia-se na fase de recrutamento. Nastiti e Setyohadi (2020) afirmam que descrições de trabalho pouco claras são uma das principais causas de contratações incorretas visto que o perfil de candidato atraído é pouco aderente às reais necessidades do projeto. Portanto, cabe ao gestor garantir descrição de trabalho adequadas.

Referente ao processo seletivo em si Nastiti e Setyohadi (2020) reforçam que a seleção de profissionais deve ser focada tanto nos aspectos educacionais quanto na experiência dos candidatos, os quais podem ser avaliados durante entrevistas. Paredes-Valverde, Pilar Salas-Zárate et al. (2018) apresentam uma proposta de ontologia que cobre quais aspectos devem ser avaliados durante o processo seletivo.

Idealmente o time deve ser heterogêneo, significando que nenhuma pessoa deve ter a mesma expertise de outro. Para Nastiti e Setyohadi (2020), isso agrega amplitude às discussões e melhora a qualidade das decisões tomadas. Já Schloegel et al. (2016) argumentam que além de amplitude técnica o time deve possuir diversidade de idades dentre os membros.

Angelis (2019) apresenta resultados empíricos que suportam a afirmação de que traços de personalidade impactam na produtividade de times de desenvolvimento de software. O gestor do projeto pode fazer uso de testes psicométricos com apoio de profissionais da área da psicologia para identificar os perfis dos indivíduos.

O modelo *Big Five* de categorização divide os perfis humanos em cinco grandes áreas: extroversão, amabilidade, conscienciosidade, neuroticismo e abertura. Fazendo uso deste modelo, Caulo et al. (2021b) propõe que os perfis com maior grau de amabilidade e conscienciosidade correspondem às pessoas com maior produtividade em desenvolvimento de software.

Além deste, Meyer, Zimmermann e Fritz (2017) propõe em seu trabalho uma nova forma de categorização dos profissionais de desenvolvimento de software, dessa vez em seis categorias: desenvolvedor social, desenvolvedor solitário, desenvolvedor focado, desenvolvedor equilibrado, desenvolvedor líder e desenvolvedor orientado a objetivos, o autor apresenta sugestões de forma de tratamento para cada tipo de profissional.

Por fim, Fatema e Kazi Sakib (2017), Bass et al. (2018) Wang, Huang e Liu (2018) e Stylianou e Andreou (2016) ressaltam a importância de maximizar a entrega enquanto minimiza-se o *overhead* de comunicação (problema apresentado na Seção 4.1.1). Cabe ao gestor buscar encontrar o ponto de equilíbrio a partir do qual a adição de novos membros reduz a quantidade de software entregue. Em virtude da quantidade de caminhos de comunicação, um time de desenvolvimento não deve exceder 9 membros.

Analisando os estudos primários relacionados à montagem de time foram propostas quatro práticas específicas a serem validadas pelos especialistas via questionário, são elas:

- **Avaliar traços de personalidade e soft skills:** Avaliar traços de personalidade e *soft skills* garante a correta atribuição de membros a projetos, considerando as especificidades

técnicas e a capacidade dos indivíduos. Isso aumenta a produtividade e a satisfação no trabalho e pode ser implementado através de:

1. Questionários de personalidade: Questionário que consiste em perguntas para avaliar diferentes traços de personalidade, como extroversão, neuroticismo, abertura a novas experiências, amabilidade e conscienciosidade. Exemplos: Big Five Inventory (BFI) e Myers-Briggs Type Indicator (MBTI). A aplicação de testes psicométricos requer o acompanhamento e interpretação de um profissional treinado como psicólogo ou outros profissionais especialistas;
 2. Entrevistas comportamentais estruturadas: Entrevistas projetadas para avaliar como um candidato se comportaria em situações hipotéticas em que o candidato é convidado a responder com base em sua experiência prévia. As respostas ajudam a avaliar traços como liderança, trabalho em equipe e resolução de problemas;
 3. Testes de desempenho: Testes que consistem na realização de tarefas específicas como escrever código de computador. A forma com que o candidato aborda a tarefa fornece informações sobre a capacidade de trabalhar sob pressão ou atenção aos detalhes;
 4. Observação direta: Observar o comportamento do profissional em situações reais.
- **Montar time heterogêneo:** Formar um time heterogêneo permite que a abordagem de problemas de diferentes maneiras e, para os quais, encontrem soluções criativas e eficazes. Isso pode ser implementado através de uma política de recrutamento e seleção que valoriza a diversidade, o que pode incluir diversidade de gênero, raça, etnia, orientação sexual, idade, entre outros.

Cabe ao gestor criar um ambiente que incentiva a colaboração entre membros da equipe e que promova a troca de ideias e conhecimentos, permitindo que os membros da equipe aprendam uns com os outros e trabalhem juntos para alcançar objetivos comuns.
 - **Criar job description para contratação:** Descrições de trabalho claras evitam contratações incorretas, garantindo a seleção de profissionais adequados para cada função. Isso economiza tempo e recursos na contratação e aumenta a satisfação do funcionário. Pode ser implementado com a ajuda de especialistas em RH e *feedback* dos membros da equipe.
 - **Definir tamanho do time:** Definir o tamanho ideal do time é uma tarefa do gestor, que deve equilibrar a quantidade de membros e a eficiência na entrega do software. O tamanho ideal de uma equipe pode depender de vários fatores, incluindo a complexidade do

projeto, a experiência dos membros da equipe e a natureza das tarefas a serem realizadas. Em geral, o aumento do tamanho da equipe pode aumentar a eficiência até certa medida, mas, depois disso, o custo de coordenação aumenta e a eficiência começa a diminuir. O custo de coordenação cresce exponencialmente na proporção que os caminhos de comunicação entre os membros do time aumenta.

4.2.2 Treinamento

De acordo com o identificado por Shahzad, Xiu e Shahbaz (2017), os colaboradores esperam que a organização esteja comprometida em prover constante treinamento aos seus profissionais, enquanto Bass et al. (2018) complementa que o gestor deve desafiar os membros do time a aprender novas habilidade.

O primeiro passo para garantir treinamento ao time é um efetivo gerenciamento de competências segundo Song et al. (2020) e Angelis (2019). Com isso pode-se identificar lacunas de conhecimento e profissionais que podem se tornar eventuais disseminadores técnicos. Os treinamentos não devem limitar-se às *hard skills*. Cardenas-Castro, Julio e Rodriguez (2019) argumentam que treinamentos psicológicos melhoram as habilidades humanas e bem estar dentro do time, seus resultados são mais úteis nos estágios iniciais do projeto do que quando em comparação com estágios de software em produção.

Nicolaescu et al. (2020) sugerem o uso de treinamentos em conjunto com acompanhamento de carreira e plano de sucessão, mantendo a motivação dos profissionais.

Analisando os estudos primários relacionados à treinamento foram propostas quatro práticas específicas a serem validadas pelos especialistas via questionário, são elas:

- **Desafiar time a aprender novas habilidades:** O gestor deve incentivar o time a adquirir novas habilidades, promovendo o crescimento profissional e a adaptabilidade da equipe. Isso aumenta a competitividade e a flexibilidade da equipe. Pode ser implementado através de treinamentos, *workshops* e incentivos para a educação contínua.
- **Identificar e gerenciar competências do time:** Mapear os conhecimentos necessários e os existentes no time permite um gerenciamento efetivo das competências, contribuindo para a eficiência e o desenvolvimento do projeto. Essa prática pode ser implementada por meio de avaliações de competência de forma periódica e planos de treinamento personalizados.

- **Organizar treinamentos:** Fornecer treinamento constante aos profissionais melhora suas habilidades e aumenta a eficiência do projeto. Isso pode ser implementado por meio de programas de treinamento internos ou externos e aprendizado online.

Recomendam-se treinamentos compostos por uma combinação de treinamento formal, aprendizado baseado em trabalho real e mentoria de desenvolvedores mais experientes.

Profissionais sênior podem ser uma boa opção para ministrar treinamentos, mas não necessariamente a única. A empresa pode incentivar todos seus profissionais a se tornarem mentores, oferecendo oportunidades de desenvolvimento de carreira e reconhecimento por seu trabalho. A empresa também pode fornecer treinamento e suporte para os mentores, para que eles possam desempenhar suas funções com eficácia.

A criação de um ambiente de trabalho que valorize a aprendizagem e o desenvolvimento contínuo incentivam as pessoas a participarem dos treinamentos, e é comum que as empresas emitam certificados para os funcionários que concluem os programas de treinamento. Isso pode ajudar a reconhecer o esforço e o desempenho dos funcionários, além de fornecer uma prova tangível de suas habilidades e conhecimentos adquiridos. No entanto, a decisão de emitir certificados após a conclusão do treinamento depende das políticas e práticas da empresa.

- **Montar plano de carreira e plano de sucessão:** A implementação de planos de carreira e sucessão, aliados a treinamentos, ajuda a motivar e reter talentos, além de garantir a continuidade do projeto quando um profissional deixa a empresa ou muda de posição.

Essa prática pode ser implementada por meio de reuniões de carreira individuais e planos de desenvolvimento personalizados. A prática inclui: identificar potenciais sucessores, fornecer treinamento, oferecer oportunidades de mentoria, criação de plano de transição e realização de testes de sucessão.

É recomendado que a empresa identifique mais de um potencial sucessor para cada posição crítica, para garantir que haja uma opção de backup caso o primeiro candidato não esteja disponível ou não seja adequado para a posição. Além disso, é importante que a empresa avalie regularmente seus planos de sucessão e faça ajustes conforme necessário, para garantir que estejam atualizados e alinhados com as necessidades da empresa e dos funcionários.

4.2.3 Comunicação

Os impactos de uma comunicação ineficiente são amplamente discutidos na literatura, entretanto, pouco é encontrado no que tange a boas práticas para garantir boa comunicação.

A disseminação de informações referentes ao progresso do desenvolvimento de software é uma atividade recorrente de comunicação. Bass (2016) sugere o emprego de ferramentas on-line para representação virtual de quadros Kanban, esta sendo uma prática altamente relacionada com as práticas de monitoramento (Seção 4.2.8)

Quanto a periodicidade da comunicação Margareth e Mulyanto (2021) indicam que a frequência da comunicação impacta positivamente na produtividade. Portanto, o time deve manter uma comunicação fluída e evitando protocolos fixos de comunicação entre membros do time e entidades externas, acontecendo de forma natural.

"*Feedback* é a informação provida por um agente a respeito da performance ou entendimento de alguém" (Hattie e Timperley 2007). Zaouga, Rabai e Alalyani (2019) caracterizam a atividade de prover feedback como parte do processo de gerenciar o time do projeto.

De acordo com Bass et al. (2018) a prática de feedback é considerada por desenvolvedores de software como um forma de reconhecimento, e a falta de feedback é um problema de comunicação que impacta diretamente na motivação e, por consequência, na qualidade da entrega do time (Magalhães 2017).

Fatema e Kazi Sakib (2017) e Machuca-Villegas et al. (2022) concordam que feedback adequado está entre os principais fatores que influenciam a produtividade de times de desenvolvimento de software.

Analisando os estudos primários relacionados à comunicação foram propostas três práticas específicas a serem validadas pelos especialistas via questionário, são elas:

- **Prover feedback ao time:** Prover *feedback* faz parte do gerenciamento efetivo do time de projeto, auxiliando na melhoria contínua dos membros da equipe, aumenta a moral da equipe e a qualidade do trabalho. É importante a existência de um ambiente de confiança e respeito mútuo, em que o *feedback* seja visto como oportunidade de crescimento e aprendizado, em vez de crítica pessoal. Pode ser implementado através de reuniões de *feedback* regulares e sistemas de avaliação entre pares.

Algumas técnicas comuns para estruturar um *feedback* adequado incluem: ser específico e objetivo, fornecer exemplos concretos, enfatizar o comportamento em vez da personali-

dade, usar uma linguagem clara e direta, e fornecer sugestões construtivas para melhorias futuras.

Qualquer membro da equipe pode fornecer *feedback*, desde que seja feito de forma construtiva e respeitosa, com o objetivo de ajudar a equipe a melhorar.

- **Praticar gestão à vista:** A disponibilização de indicadores de maneira visível para todo o time, seja em quadro físico ou ferramenta online, promove a transparência e o alinhamento das metas. Isso aumenta a consciência da equipe sobre o status do projeto e permite que o próprio time identifique rapidamente os problemas e as áreas que precisam de mais atenção, permitindo que eles sejam abordados de forma proativa antes que se tornem um obstáculo para o progresso do projeto.

Pode ser implementado com o uso de quadros de gestão à vista, dashboards online e reuniões de status regulares.

Recomenda-se que as medidas sejam feitas como equipe única, para que "não possa haver vencedores ou perdedores individuais" e todos trabalhem em direção ao mesmo objetivo.

- **Criar canais e promover comunicação aberta:** A manutenção de uma comunicação fluida e o estímulo a evitar protocolos fixos contribuem para o fluxo de informação natural e produtivo entre os membros da equipe. Isso reduz mal-entendidos e aumenta a eficiência da equipe. Pode ser implementado através de reuniões regulares, uso de ferramentas de comunicação online e promoção de um ambiente aberto e inclusivo.

4.2.4 Onboarding

O processo de *onboarding* ou entrada de novos profissionais em um time de desenvolvimento deve ser construído de forma a otimizar a passagem de conhecimento aos novos membros. Britto et al. (2019) sugerem como prática mapear todos os conhecimentos que são esperados que um novo profissional absorva juntamente com os responsáveis por essa passagem de conhecimento.

O processo de entrada de novos membros a times de projetos distribuídos globalmente pode levar mais tempo quando comparado ao tempo necessário para *onboarding* em times na mesma localização. Britto et al. (2019) citam como exemplo projetos de alta complexidade e código legado, para os quais mentoria remota durante o período de 4-6 meses pode ser insuficiente. O autor sugere que deve ser priorizado o emprego de mentores que possam fazer

a passagem de conhecimento de forma presencial e síncrona para aumentar a velocidade do progresso do time, além de buscar mentores com boas habilidades de comunicação.

Analisando os estudos primários relacionados a onboarding foram propostas duas práticas específicas a serem validadas pelos especialistas via questionário, são elas:

- **Identificar e gerenciar competências do time:** Mapear os conhecimentos necessários e os existentes no time permite um gerenciamento efetivo das competências, contribuindo para a eficiência e o desenvolvimento do projeto. Essa prática pode ser implementada por meio de avaliações de competência de forma periódica e planos de treinamento personalizados.
- **Executar checklist de onboarding:** A criação de *checklists* para *onboarding* auxilia na padronização e garantia da completa integração de novos membros ao time, reduzindo o tempo de aclimatação e aumentando a eficiência. Essa prática pode ser implementada com a criação de um *checklist* que inclua todas as etapas necessárias para a integração de um novo membro.

4.2.5 Emprego de práticas ágeis

Machuca-Villegas et al. (2022) afirmam que os membros do time devem ser empoderados a tomar decisões, organizando-se para estabelecer e alcançar objetivos. Para alcançar maturidade de auto-organização relacionada à atribuição, autonomia e definição de métodos de trabalho, Magalhães (2017) sugere o emprego de métodos como XP e Scrum.

Em organizações cujos times não empregam tais métodos ágeis e há forte dependência do papel do gestor, Shahzad, Xiu e Shahbaz (2017) sugerem que este questione constantemente o time se há uma forma melhor de fazer as coisas.

Programação em par é uma prática ágil que consiste em dois programadores compartilhando uma mesma estação de trabalho, um escreve o código enquanto o outro simultaneamente analisa o trabalho realizado. Caulo et al. (2021b) e Fatema e Kazi Sakib (2017) listam programação em par como prática percebida como relacionada a aumento de produtividade.

Analisando os estudos primários relacionados ao emprego de práticas ágeis foi proposta uma prática específicas a ser validada pelos especialistas via questionário, sendo:

- **Usar práticas ágeis:** O emprego de métodos ágeis, como XP e Scrum, promove a eficiência e a adaptabilidade do projeto. A implementação dessas práticas pode ser feita

através de treinamentos em métodos ágeis para a adoção de uma mentalidade ágil.

4.2.6 Otimização de alocação

Durante a etapa de planejamento de projetos de software, o gerente de projeto deve resolver um "quebra-cabeças": o problema de alocação em projetos de software. Sobre isso, a literatura recente apresenta múltiplas propostas de solução otimizada respeitando diferentes restrições.

Objetivando a criação de um cronograma mínimo Vega-Velazquez, Garcia-Najera e Cervantes (2018) apresenta modelos que equilibram tempo e custos. Chiang e Lin (2020) agregam a seu modelo as habilidades do pessoal envolvido para que as estimativas sejam mais factíveis. Nigar (2017) introduz um modelo matemático que considera cinco fatores: duração do projeto, fragmentação de tarefas, robustez, custo e estabilidade.

Apresentando-se como alternativa aos modelos que consideram um número limitado de restrições, X. Shen, Guo e Li (2020) apresentam alternativas para aumento do número de variáveis consideradas e X.-N. Shen et al. (2018) propõe um algoritmo baseado em aprendizado de máquina para propor alocação de indivíduos de forma dinâmica, considerando aspectos como capacidade e motivação para aprender e evolução de proficiência técnica ao longo do tempo.

Por fim, foi encontrado na literatura proposta de modelo específico para otimização da alocação de recursos considerando a cultura corporativa única das companhias estatais chinesas (Song et al. 2020).

Analisando os estudos primários relacionados a otimização de alocação foi proposta uma prática específicas a ser validada pelos especialistas via questionário, sendo:

- **Criar cronograma mínimo:** A criação de um cronograma mínimo equilibra tempo e custo, auxiliando na execução eficaz do projeto. Isso ajuda a equipe a se concentrar nas tarefas mais importantes e a evitar atrasos.

O planejamento antecipado da alocação dos profissionais auxilia na definição da composição do time, minimizando a necessidade de mudanças no decorrer do projeto. A alocação do recurso correto para a atividade pode considerar fatores individuais (interesse pessoal, atitude, confiança, experiência, conhecimento, papel desempenhado) e critérios do projeto (tipo de projeto, custo, criticidade, risco, complexidade).

4.2.7 Job rotation

Antes da década de 1950 a vasta maioria das estratégias de trabalho em indústrias tradicionais eram caracterizadas por simplificação, especialização e repetição. Atualmente, em indústrias de conhecimento, como a área de desenvolvimento de software, estas mesmas características são reconhecidas por gerar monotonia, tédio, fadiga e, conseqüentemente, diminuição da performance dos indivíduos (Magalhães 2017).

Job rotation é a prática de mover indivíduos de projetos de software dentro de uma mesma organização. Santos (2017) caracteriza as movimentações em dois formatos:

- **Job-to-job:** no qual o profissional muda de responsabilidades e passando a desempenhar uma nova função dentro da organização. Ex: um desenvolvedor passa a desempenhar as funções de um analista de qualidade;
- **Project-to-project:** cujo foco é mover o profissional para outro projeto e/ou time mantendo as mesmas atribuições de antes da rotação.

Franca, Silva and Sharp (2020) ressaltam que esta pode ser uma prática positiva em termos de motivação e satisfação dos indivíduos, fazendo com que os profissionais estejam sempre desafiados a aprender novas habilidades.

Em contraponto, Kula et al. (2021) indicam que para entregas dentro do prazo e otimização de performance os gestores devem tentar manter times estáveis, em que os membros possam construir familiaridade de longo termo entre si. Portanto, cabe ao gestor encontrar o ponto de equilíbrio entre motivação gerada por novas oportunidades e otimização de performance oriunda da manutenção de um time estável.

Analisando os estudos primários relacionados a job rotation foi proposta uma prática específicas a ser validada pelos especialistas via questionário, sendo:

- **Rotacionar time:** A rotação de time pode ser realizada de duas maneiras:
 1. A rotação de tarefas (*job-to-job*) dentro do mesmo time ou projeto permite aos membros adquirirem uma variedade de habilidades e conhecimentos através da mudança de papéis entre membros do mesmo time. Pode ser implementada através de um programa de rotação de tarefas bem estruturado.
 2. A rotação entre projetos (*project-to-project*) permite aos membros trazerem novas perspectivas e aprender com diferentes contextos. Pode ser implementado ao designar

membros da equipe para novos projetos em um ciclo regular, o que também ajuda a evitar a estagnação e promover a inovação.

A frequência ideal da rotação de trabalho pode variar dependendo do contexto específico e das necessidades da organização, ela deve ser frequente o suficiente para fornecer aos funcionários novos desafios e oportunidades de aprendizado, mas não tão frequente que interrompa o trabalho ou cause instabilidade.

4.2.8 Monitoramento de performance

Medir e monitorar performance em desenvolvimento de software, de acordo com Nicolaescu et al. (2020) é uma atividade chave para gestores. Oliveira et al. (2016) argumentam que as medidas podem ser utilizadas para comparar eficiência de diferentes desenvolvedores na mesma organização ajudando a compor e reajustar os times.

A posse de métricas de desempenho, de acordo com Zaouga, Rabai e Alalyani (2019) oportuniza *feedbacks* de boas qualidade (vide Seção 4.2.3) visando a melhoria de performance dos profissionais.

Nastiti e Setyohadi (2020) sugerem que a medição de performance pode ser feita através de KPI's ou *Key Performance Indicators*, ou seja, indicadores-chave de performance acordados previamente com o próprio time. Tradicionalmente performance é relacionada com eficiência e produtividade. Cardenas-Castro, Julio e Rodriguez (2019) argumentam que além destes, é essencial que sejam acordados indicadores de performance que avaliem também *soft skills* individuais, promovendo assim mais colaboração.

Os indicadores podem ser disponibilizados em quadro ou ferramenta online visível para todo o time, dessa forma, um time que tenha sido treinado em interpretar os indicadores pode fazer monitoramento mútuo de performance (Fatema e Kazi Sakib 2017).

Analisando os estudos primários relacionados a monitoramento de performance foram propostas duas práticas específicas a serem validadas pelos especialistas via questionário, são elas:

- **Praticar gestão à vista:** A disponibilização de indicadores de maneira visível para todo o time, seja em quadro físico ou ferramenta online, promove a transparência e o alinhamento das metas. Isso aumenta a consciência da equipe sobre o status do projeto e permite que o próprio time identifique rapidamente os problemas e as áreas que precisam de mais atenção, permitindo que eles sejam abordados de forma proativa antes que se tornem um

obstáculo para o progresso do projeto.

Pode ser implementado com o uso de quadros de gestão à vista, dashboards online e reuniões de status regulares.

Recomenda-se que as medidas sejam feitas como equipe única, para que "não possa haver vencedores ou perdedores individuais" e todos trabalhem em direção ao mesmo objetivo.

- **Usar indicadores para monitorar a performance do time de desenvolvimento de software:** O uso de indicadores-chave de desempenho ou *key performance indicators* (KPIs) permite medir e monitorar o desempenho do time, promovendo a melhoria contínua. Essa prática pode ser implementada através do uso de software de rastreamento de projetos e ferramentas de análise de desempenho.

5 VALIDAÇÃO

Este capítulo apresenta o processo seguido, dados obtidos via questionário com especialistas na área de projetos de desenvolvimento de software e discute os resultados encontrados.

A primeira seção apresenta e discute os dados demográficos dos respondentes, na segunda seção cada prática e suas respostas é discutida individualmente, enquanto na terceira seção é apresentado e discutido uma análise geral das práticas.

5.1 Dados demográficos

O questionário enviado aos especialistas possui inicialmente quatro perguntas demográficas com objetivo de identificar o perfil dos respondentes, todas de múltipla escolha, com os participantes orientados a selecionar aquela opção que se encaixe melhor à sua realidade:

- Q1: Qual é a sua área de atuação na organização em que trabalha?
- Q2: Qual a sua área de atuação?
- Q3: Qual a sua experiência profissional?
- Q4: Qual o seu maior título escolar?

A Tabela 5.1 apresenta as informações demográficas fornecidas pelos especialistas.

A primeira pergunta identifica qual a área de atuação na organização em que o respondente trabalha, possuindo quatro opções de resposta: Análise, Desenvolvimento, Gestão e Qualidade. A distribuição das respostas pode ser visualizada na Figura 5.1.

É possível observar divisão entre os respondentes, aproximadamente metade dos respondentes que atuam na gestão e a outra metade atua na parte técnica de suas empresas, dividido entre desenvolvimento, análise e qualidade. Essa divisão enriquece o resultado pois garante que analisemos os resultados sob a ótica do gestor que é responsável por aplicar as práticas e o técnico que é diretamente impactado pelas mesmas, ambos trazendo as suas percepções sobre os resultados alcançados.

Tabela 5.1 – Informações demográficas dos respondentes do questionário

# Especialista	Q1	Q2	Q3	Q4
E1	Gestão	Governo	1-10 anos	Bacharelado
E2	Desenvolvimento	Comércio	1-10 anos	Bacharelado
E3	Gestão	Academia	1-10 anos	Doutorado
E4	Desenvolvimento	Academia	21+ anos	Bacharelado
E5	Gestão	Academia	21+ anos	Mestrado
E6	Gestão	Academia	11-20 anos	Mestrado
E7	Gestão	Serviços	21+ anos	Doutorado
E8	Gestão	Indústria	11-20 anos	Doutorado
E9	Análise	Comércio	11-20 anos	Bacharelado
E10	Desenvolvimento	Serviços	11-20 anos	Mestrado
E11	Desenvolvimento	Indústria	11-20 anos	Doutorado
E12	Gestão	Academia	21+ anos	Especialização
E13	Gestão	Academia	11-20 anos	Bacharelado
E14	Desenvolvimento	Serviços	11-20 anos	Especialização
E15	Análise	Indústria	11-20 anos	Bacharelado
E16	Gestão	Indústria	21+ anos	Mestrado
E17	Gestão	Academia	21+ anos	Mestrado
E18	Gestão	Serviços	11-20 anos	Bacharelado
E19	Gestão	Serviços	1-10 anos	Especialização
E20	Desenvolvimento	Serviços	11-20 anos	Bacharelado
E21	Análise	Serviços	1-10 anos	Bacharelado
E22	Qualidade	Indústria	11-20 anos	Bacharelado
E23	Desenvolvimento	Serviços	1-10 anos	Bacharelado
E24	Análise	Indústria	1-10 anos	Ensino médio
E25	Desenvolvimento	Serviços	1-10 anos	Bacharelado
E26	Gestão	Serviços	1-10 anos	Especialização
E27	Gestão	Serviços	1-10 anos	Bacharelado
E28	Qualidade	Serviços	1-10 anos	Especialização
E29	Gestão	Serviços	11-20 anos	Especialização
E30	Gestão	Serviços	1-10 anos	Bacharelado
E31	Gestão	Serviços	11-20 anos	Bacharelado

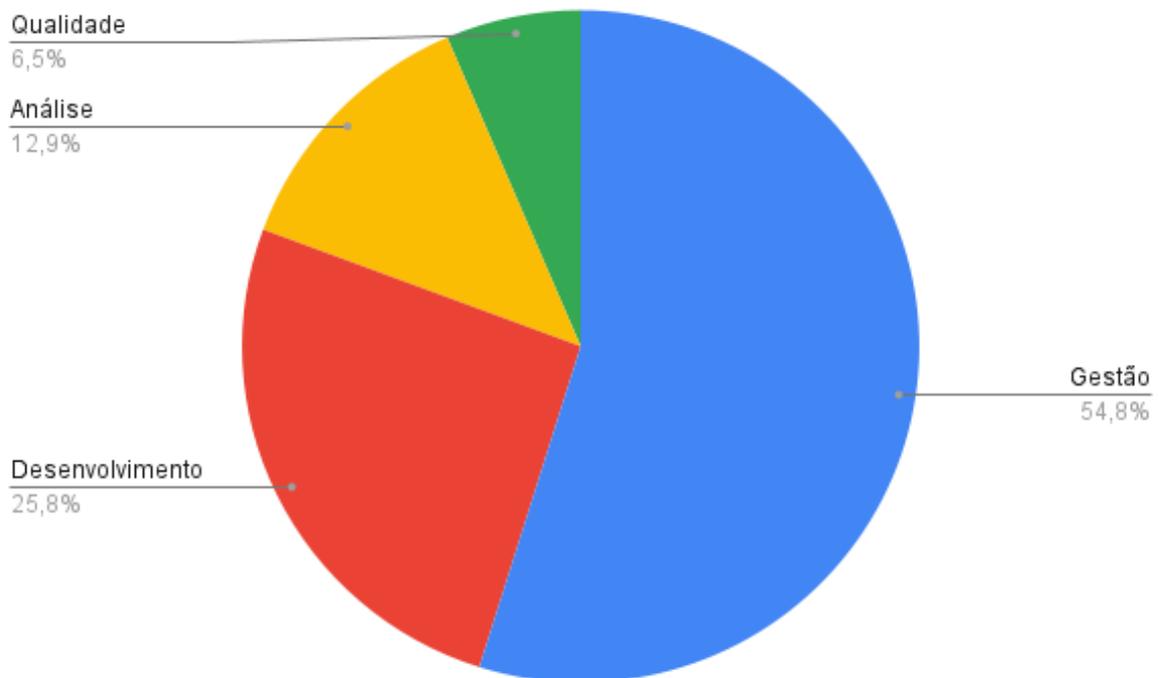


Figura 5.1 – Qual é a sua área de atuação na organização em que trabalha?

A segunda pergunta identifica a área de atuação da organização que o respondente faz parte, possuindo cinco opções de resposta: Academia, Comércio, Indústria, Governo e Serviços. A distribuição das respostas pode ser visualizada na Figura 5.2.

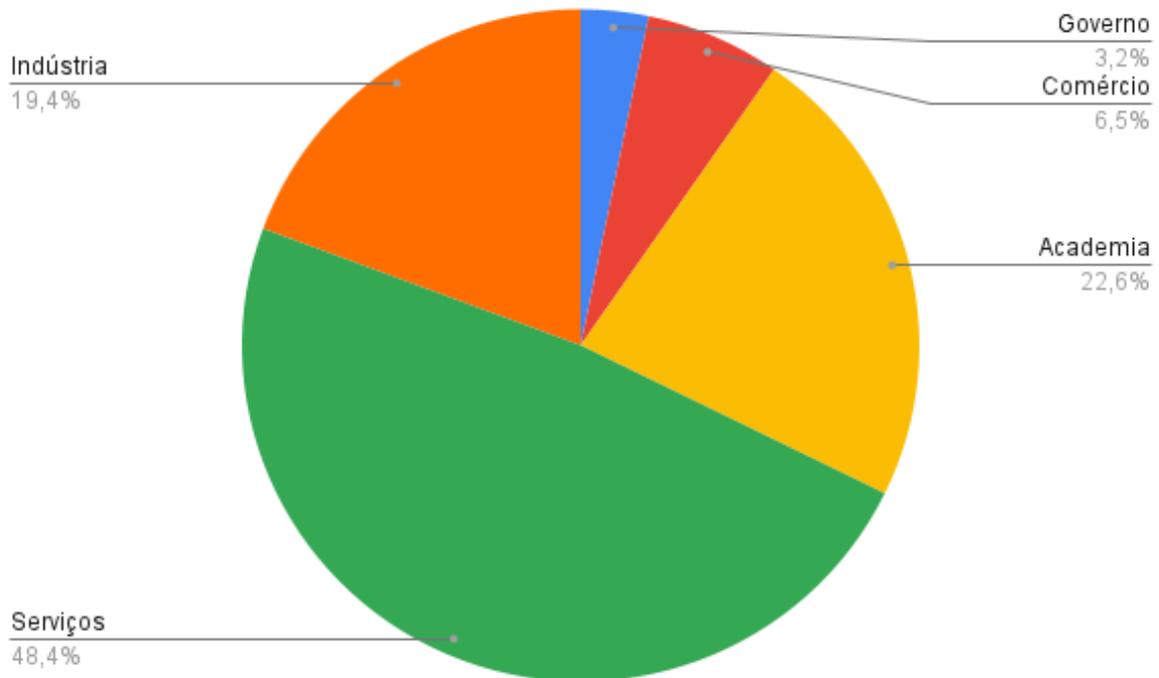


Figura 5.2 – Qual a sua área de atuação?

Há grande representatividade de profissionais que atuam na área de serviços, quase metade dos respondentes, enquanto a outra metade divide-se entre profissionais na academia e indústria. As áreas de comércio e governo tiveram poucos respondentes.

A partir desses dados é possível identificar se há diferença entre a percepção de relevância de práticas em diferentes áreas, especialmente serviços comparado com as restantes.

A terceira pergunta identifica o tempo experiência profissional dos respondentes em intervalo de anos, possuindo três opções de respostas: 1-10 anos, 11-20 anos e 21+ anos. A distribuição das respostas pode ser visualizada na Figura 5.3.

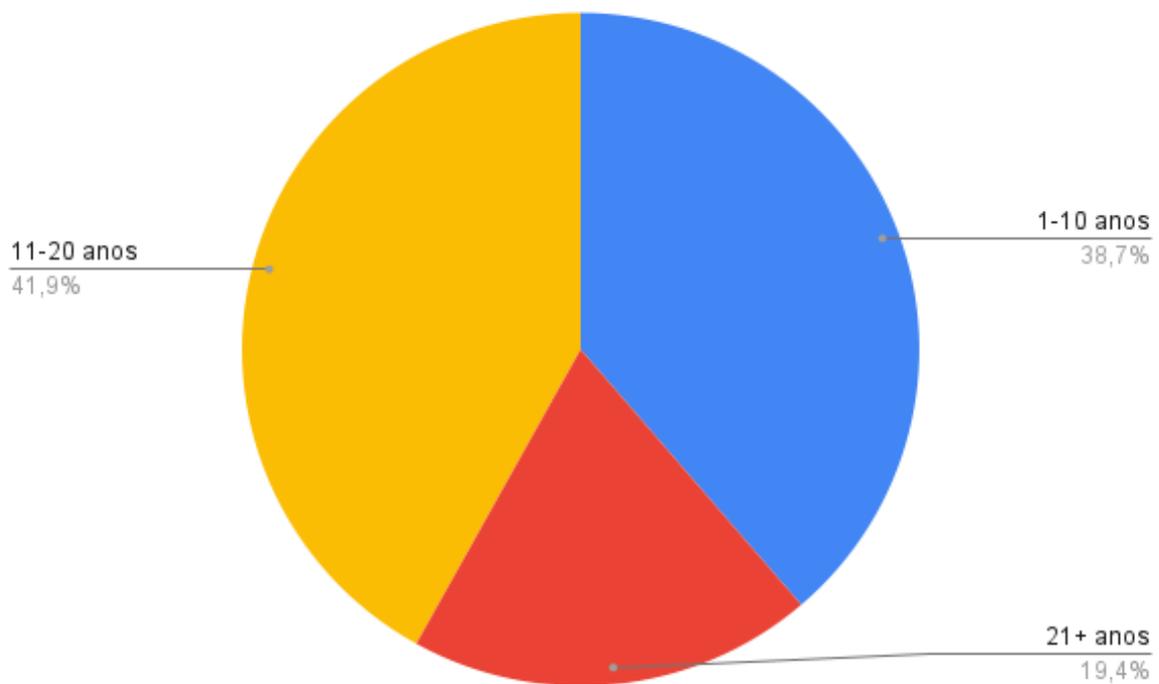


Figura 5.3 – Qual a sua experiência profissional?

Diferentemente das outras perguntas, onde pelo menos uma das opções de resposta teve um volume inferior a 10% dos respondentes, nessa pergunta todos tendo um volume superior de respostas.

A maior parte dos respondentes, 41,9% possui entre 11-20 anos de experiência, 38,7% dos respondentes estão em início de carreira, com experiência variando entre 1-10 anos e por último 19,4% dos respondentes possuem mais de 21 anos de experiência. É possível supor que seja natural a maior presença de respondentes nos dois primeiros grupos tendo em vista a recente popularização e acesso à tecnologia da informação, enquanto profissionais com mais de 21 anos de carreira são pessoas que tiveram o primeiro acesso a computadores na década de 1990 ou antes, um universo mais restrito da realidade brasileira.

A quarta pergunta identifica a maior titulação escolar dos respondentes, possuindo cinco opções de resposta: Ensino médio, Bacharelado, Especialização, Mestrado e Doutorado. A distribuição das respostas pode ser visualizada na Figura 5.4.

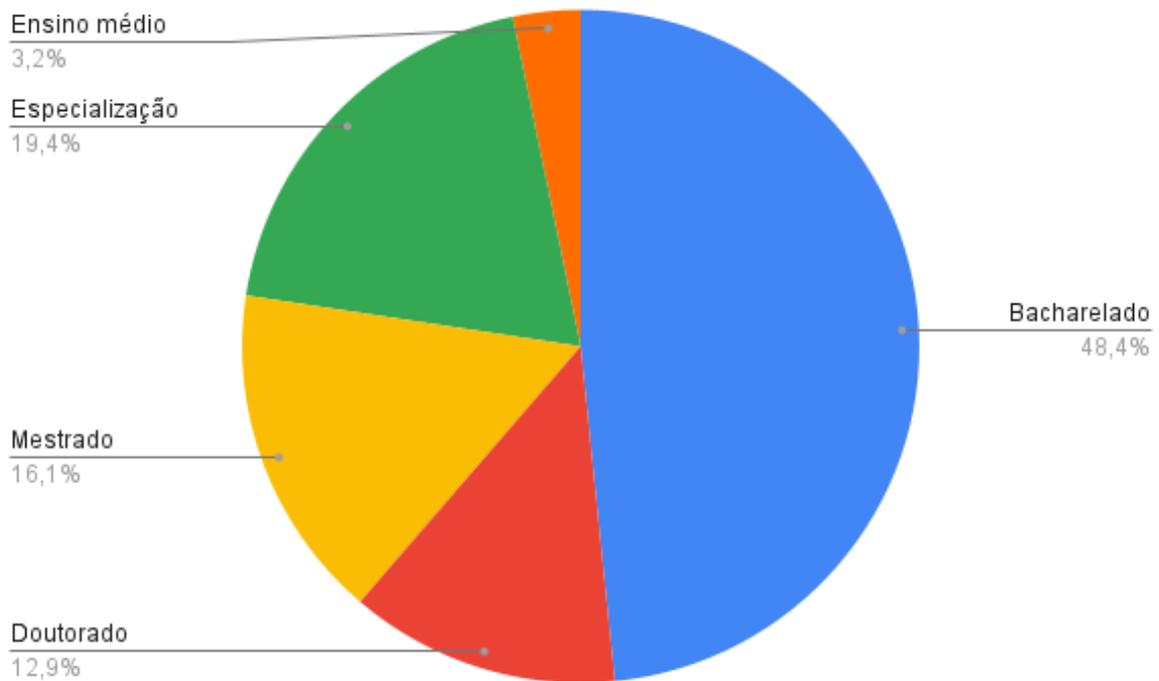


Figura 5.4 – Qual o seu maior título escolar?

Aproximadamente metade dos respondentes possuem um título de bacharel, enquanto a outra metade divide-se em títulos de pós-graduação: 19,4% possuem especialização, 16,1% mestrado e 12,9% doutorado.

Dentro da amostra analisada 96,8% dos respondentes possuem ensino superior, enquanto 3,2% possuem ensino médio. Visto que legalmente não há obrigatoriedade por uma titulação mínima para atuar em projetos de desenvolvimento de software, nenhum dado foi descartado.

5.2 Práticas

Esta seção apresenta e discute das percepções dos especialistas em relação à importância das práticas identificadas para a validação através do questionário, seguindo a metodologia estabelecida.

A Tabela 5.2 contém as respostas de todos os especialistas referente a relevância geral das práticas, questões identificadas por P0XX, e questões referentes a relevância da prática para resolução de um problema alvo específico, questões identificadas por P0XXAY, onde XX é o identificador numérico da prática e Y é o identificador numérico do problema alvo.

Tabela 5.2 – Respostas dos participantes sobre relevância a das práticas

Pergunta	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13	E14	E15	E16	E17	E18	E19	E20	E21	E22	E23	E24	E25	E26	E27	E28	E29	E30	E31		
P001	3	2	3	0	3	3	3	0	3	3	3	2	3	1	2	3	3	2	2	3	3	2	3	3	3	2	3	2	3	2	3	3	
P001A1	2	3	2	2	2	2	2	3	2	2	2	2	2	2	3	2	2	2	2	2	3	3	2	2	3	3	3	3	3	3	3	3	
P001A2	2	2	1	1	3	2	2	3	2	2	3	1	2	1	2	3	2	1	2	1	2	1	1	3	1	3	2	3	2	3	2	3	
P001A3	3	2	2	3	2	2	2	3	2	3	2	2	2	2	2	2	2	2	2	2	2	3	1	3	1	3	2	2	2	3	2		
P001A4	2	2	1	2	3	3	2	3	2	2	3	1	2	1	3	1	2	1	2	1	2	1	1	3	2	3	1	2	2	2	3	2	
P001A5	3	2	2	2	2	2	2	3	2	3	3	2	3	3	3	3	2	2	3	2	2	3	2	2	2	2	2	2	2	2	3	3	
P001A6	3	3	1	3	3	2	3	3	2	3	2	2	2	2	3	1	2	1	3	1	1	3	3	2	3	2	2	2	3	2	3		
P2	3	3	3	1	2	2	2	3	2	3	2	2	2	2	3	3	3	3	3	3	3	2	3	3	2	2	3	2	3	3	3		
P002A1	3	2	3	0	2	1	2	1	2	3	2	2	1	1	2	2	3	2	2	2	2	1	3	3	2	2	3	1	2	3	2	2	
P002A2	2	2	3	1	2	1	2	3	2	2	2	2	2	2	2	2	3	2	2	2	2	2	3	1	1	3	2	2	3	3	3	3	
P003	3	3	2	1	2	3	2	1	3	3	2	2	2	2	2	2	2	3	2	3	3	3	3	3	3	3	2	3	3	3	3	2	
P003A1	3	1	2	2	2	2	2	0	3	2	2	2	2	2	3	1	2	1	2	3	2	0	3	1	2	2	2	2	2	3	2	2	
P003A2	1	1	1	2	2	2	2	0	3	3	3	2	2	2	3	1	2	1	2	3	2	3	3	3	3	2	3	2	2	3	2	2	
P004	3	3	3	2	3	3	2	3	3	3	3	2	2	2	2	3	3	3	3	2	2	3	3	3	3	3	3	2	3	3	3	3	
P004A1	3	3	3	2	3	3	2	3	3	2	3	2	2	2	3	3	2	3	3	2	2	3	3	3	3	3	3	3	3	3	3	3	
P004A2	3	3	0	1	3	2	2	3	3	1	2	2	3	3	1	3	2	3	2	0	2	3	2	1	1	3	3	1	1	1	1	2	
P005	3	2	3	2	2	3	2	2	3	3	3	2	2	2	2	3	3	2	2	2	2	3	3	3	3	3	3	2	3	3	3	3	
P005A1	3	2	2	1	2	3	2	2	3	2	3	2	2	3	2	1	3	2	2	2	2	3	3	2	2	3	3	2	2	2	2	2	
P006	2	2	3	3	3	2	3	2	3	3	3	2	2	3	3	3	2	2	3	1	3	3	2	3	2	2	2	3	3	3	2	2	
P006A1	3	3	3	1	3	2	2	0	2	3	3	2	2	2	1	2	2	2	3	1	3	3	2	3	2	2	2	2	3	2	3	2	3
P006A2	1	3	3	2	3	2	2	0	2	3	2	2	2	3	2	2	2	1	2	1	2	3	3	3	2	3	2	3	3	3	3	3	
P007	3	3	3	3	3	3	3	3	3	3	2	3	2	2	3	3	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	
P007A1	3	3	3	2	3	3	3	3	3	2	2	3	1	2	3	3	2	3	3	2	2	3	2	3	2	3	3	3	3	3	3	3	
P007A2	2	3	3	2	3	3	2	3	3	3	3	3	2	3	3	2	3	3	3	3	3	3	3	1	2	3	3	3	3	3	3	3	
P008	2	3	3	2	3	3	1	2	3	3	3	2	2	1	2	3	3	2	3	2	3	2	3	3	2	3	3	2	2	2	3	3	
P008A1	1	2	3	2	3	3	1	3	3	2	2	2	2	2	2	1	2	2	2	0	1	3	3	1	0	1	3	3	3	3	2	2	
P008A2	1	2	3	2	3	3	1	3	3	3	3	2	1	2	2	3	2	1	2	2	3	3	2	2	2	3	2	3	3	3	3	2	
P009	2	3	2	1	3	3	2	2	3	2	3	2	2	2	2	3	2	2	3	2	2	3	2	3	2	2	2	3	3	3	3	2	
P009A1	3	3	3	2	2	3	2	2	3	2	3	2	3	2	3	3	2	3	3	1	3	3	2	3	1	3	2	2	3	3	3	3	
P009A2	1	3	3	2	2	3	2	3	2	3	2	2	2	1	3	3	2	3	2	2	3	3	2	3	2	2	2	2	2	2	3	2	
P009A3	3	2	3	3	2	3	2	3	3	2	3	2	2	3	3	3	2	3	3	3	2	3	3	3	3	2	3	1	2	3	3	3	
P010	3	2	2	3	3	2	3	3	3	3	3	2	2	2	2	3	3	3	3	3	3	1	3	3	1	3	3	3	2	2	3	3	
P010A1	3	2	2	2	2	2	2	3	3	2	3	2	3	3	3	3	3	2	3	3	2	3	3	1	2	3	3	2	3	3	3	3	
P010A2	2	2	0	1	3	3	2	1	2	1	2	1	2	3	1	2	2	2	3	1	1	1	2	0	1	1	3	1	1	1	3	3	
P010A3	3	3	2	2	3	2	2	3	3	2	3	2	2	2	3	2	3	2	3	3	2	3	2	2	1	3	2	2	1	3	3	3	
P010A4	3	3	1	2	2	2	2	3	3	2	3	2	2	2	2	3	2	2	3	1	2	3	3	1	2	3	2	3	1	3	3	3	
P010A5	1	2	1	2	3	2	2	3	3	3	3	2	2	2	3	3	2	1	3	3	1	1	3	2	2	2	2	1	2	3	2	2	
P011	3	3	3	3	3	3	2	3	3	3	2	2	2	2	2	2	2	2	2	2	1	3	2	3	2	2	2	3	1	3	2	2	
P011A1	3	3	3	3	3	2	2	2	3	2	2	2	2	2	2	3	3	3	2	2	3	3	2	2	3	2	2	3	3	3	3	3	
P012	3	3	3	3	3	3	2	3	3	3	2	2	2	2	2	2	2	2	2	2	1	3	2	3	2	2	2	3	1	3	2	2	
P012A1	3	2	2	3	2	3	2	2	3	2	3	2	2	2	2	2	2	3	2	2	1	2	2	3	2	3	2	3	1	3	2	2	
P012A2	3	3	3	2	3	3	2	2	3	3	3	3	3	3	3	2	2	2	3	3	2	3	3	3	3	3	3	3	1	3	3	3	
P013	3	2	3	0	0	3	1	3	3	3	2	2	2	3	2	2	2	3	2	0	2	3	2	3	2	2	3	2	3	2	3	3	
P013A1	3	2	2	2	0	3	0	3	3	3	2	2	2	2	1	1	2	3	2	0	2	3	3	3	2	1	2	2	2	2	2	3	
P013A2	2	1	1	2	0	3	0	3	3	2	2	2	2	3	1	1	2	2	2	0	0	0	2	3	1	1	2	1	2	2	2	3	
P014	3	2	2	1	2	3	2	3	3	2	3	2	2	3	2	2	3	2	2	2	2	3	2	2	2	2	3	3	3	3	3	3	
P014A1	3	2	2	3	2	3	2	3	3	2	3	3	2	3	2	2	3	3	3	2	2	3	3	2	2	3	3	2	2	3	3	3	
P015	2	2	2	0	1	3	2	2	3	2	3	1	2	2	2	2	1	2	1	3	2	3	2	1	2	3	3	2	3	3	2	2	
P015A1	1	1	1	2	1	2	2	1	2	1	2	1	1	1	2	2	1	1	1	3	2	2	2	1	2	1	2	1	1	1	3	2	2
P015A2	2	2	2	3	1	3	2	3	2	1	3	1	2	2	3	3	1	2	2	3	3	3	3	2	3	3	3	3	3	3	3	3	3
P015A3	2	1	3	2	1	3	2	2	2	1	3	1	2	1	2	2	1	2	1	3	2	1	2	1	1	3	3	3	3	3	3	2	2
P016	3	2	2	1	2	2	3	3	3	2	3	1	2	2	2	2	3	2	2	2	3	3	2	2	2	3	3	2	2	3	3	3	
P016A1	1	1	1	2	2	2	2	3	3	2	3	1	2	2	2	2	3	1	2	1	2	3	2	1	2	2	3	3	2	3	2	2	
P016A2	2	1	1	2	2	2	2	3	3	2	3	1	2	1	3	2	3	1	2	2	2	3	2	1	1	3	3	2	2	1	1	2	2
P016A3	3	2	2	2	2	2	2	3	3	2	2	1	2	2	3	3	3	2	2	2	2	3	2	2	1	3	3	2	1	3	3	3	

5.2.1 Prática 001 - Usar práticas ágeis

Sobre a relevância do uso de práticas ágeis no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.5:

- 2 especialistas (6,25%) consideram a prática como "não relevante";
- 1 especialista (3,12%) considera a prática como "pouco relevante";
- 8 especialistas (25%) consideram a prática como "muito relevante";
- 20 especialistas (62,5%) consideram a prática como "absolutamente relevante".

A maioria dos especialistas considera a prática de "Usar práticas ágeis" como "muito relevante" ou "absolutamente relevante", o que indica um forte entendimento dos profissionais quanto à importância das metodologias ágeis na gestão de pessoas em projetos de desenvolvimento de software. Isso pode refletir a natureza colaborativa, adaptável e orientada a feedbacks das metodologias ágeis, que podem beneficiar a gestão de equipes e a dinâmica entre os membros.

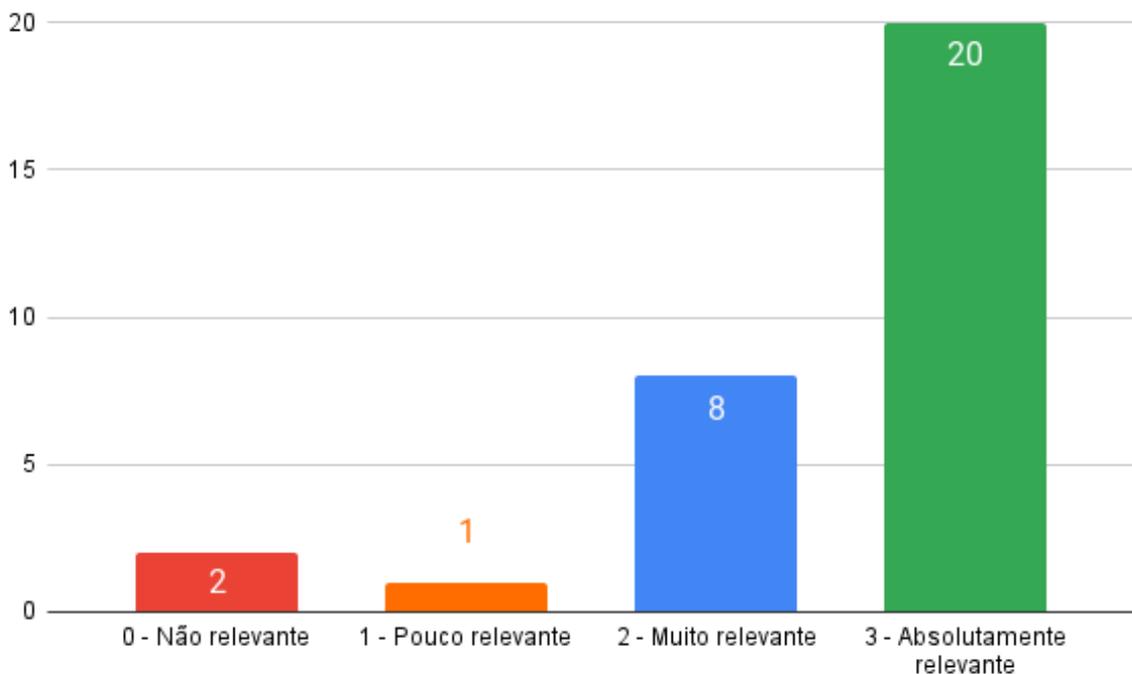


Figura 5.5 – Qual a relevância da prática 001 - Usar práticas ágeis para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 82,80%. Esse valor indica uma percepção predominantemente positiva em relação à importância de se adotar práticas ágeis para gestão de pessoas em projetos de desenvolvimento de software. Contudo, vale ressaltar que existe um pequeno grupo que não vê essa prática como relevante, o que poderia indicar a presença de outros fatores ou contextos onde as práticas ágeis podem não ser tão aplicáveis ou eficazes.

Os comentários adicionais fornecidos por dois dos especialistas oferecem perspectivas adicionais:

Muitas vezes as práticas ágeis no ramo em que estou inserido tem seu impedimentos, pois as demandas são na maioria das vezes imprevistas e não planejadas. (Especialista E14)

Ao analisar as áreas de problema alvo – Comunicação, Motivação, Comprometimento, Satisfação, Foco e Autonomia – não foi identificada uma correlação direta entre o comentário do especialista e os problemas alvo especificados no contexto da gestão de pessoas em projetos de desenvolvimento de software.

Vejo que o uso das práticas ágeis apenas pelo uso não tem tanto impacto. Na minha visão, a comunicação é a área que mais se beneficia diretamente quando as práticas são usadas e o resto é fruto disso. (Especialista E28)

A comunicação é identificada como a área que mais beneficia diretamente do uso de práticas ágeis. Isso enfatiza a ideia de que, no coração das práticas ágeis, está a colaboração e a comunicação eficaz entre os membros da equipe e as partes interessadas. Tal afirmação é suportada pelas respostas dos demais especialistas, a Figura 5.6 destaca a relevância da prática para os problemas relacionados ao tema de comunicação.

Comunicação e Foco possuem os maiores percentuais de relevância (79,57% em ambos os casos), ressaltando o papel vital que práticas ágeis desempenham em garantir que as equipes estejam alinhadas e concentradas nas prioridades certas.

Autonomia e Comprometimento também receberam percentuais elevados de relevância, com 76,43% e 73,12%, respectivamente, fruto do foco em empoderar equipes a tomar decisões e assumir responsabilidade pelo seu trabalho.

Satisfação (64,52%) e Motivação (56,59%) receberam percentuais menores. Isso pode indicar que enquanto práticas ágeis podem contribuir para a satisfação e motivação da equipe, outras práticas podem apresentar melhores resultados nessas áreas.

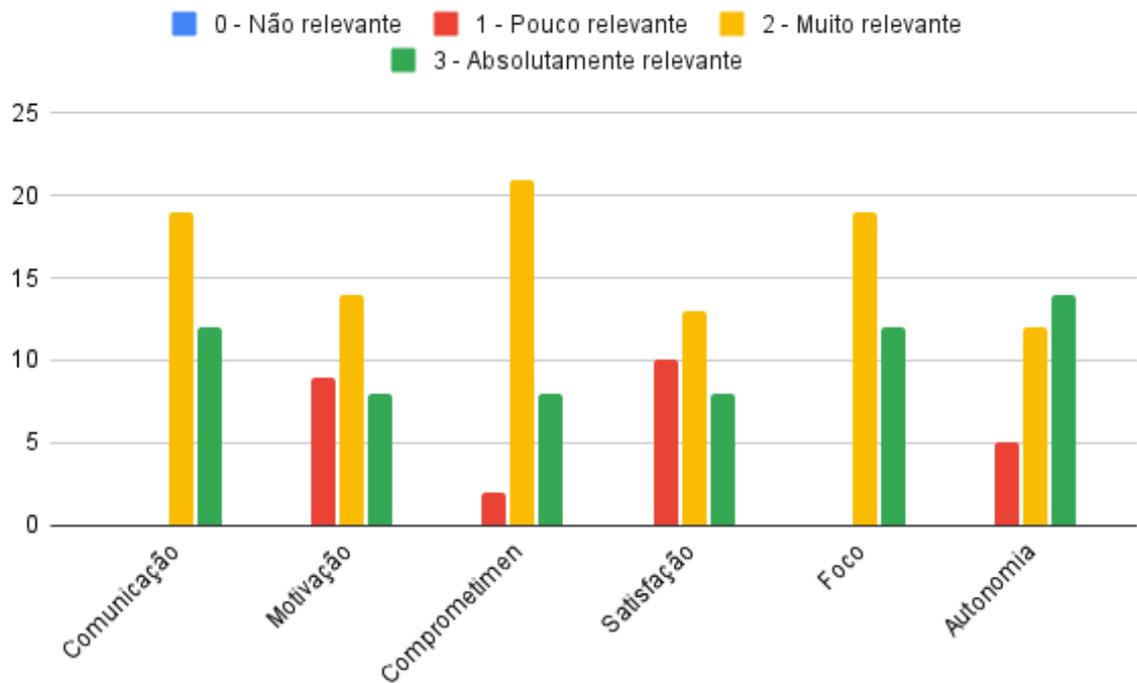


Figura 5.6 – Qual a relevância da prática 001 - Usar práticas ágeis para resolver os problemas-alvo identificados?

Considerando os especialistas que consideraram a prática como não relevante, não foi possível identificar um perfil específico em termos de área de atuação, experiência profissional ou formação acadêmica que possa sugerir um contexto específico onde estas práticas sejam consideradas menos relevantes.

5.2.2 Prática 002 - Avaliar traços de personalidade e soft skills

Sobre a relevância de avaliar traços de personalidade e soft skills no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.7:

- 0 especialistas consideram a prática como "não relevante";
- 1 especialista (3,23%) considera a prática como "pouco relevante";
- 12 especialistas (38,70%) consideram a prática como "muito relevante";
- 18 especialistas (58,07%) consideram a prática como "absolutamente relevante".

A maioria dos especialistas considera a prática de avaliar traços de personalidade e soft skills como "muito relevante" ou "absolutamente relevante", o emprego da prática auxilia o gestor na definição dos membros do time, na atribuição de tarefas de acordo com o perfil dos indivíduos e a definir a forma mais adequada para relacionar-se com cada pessoa.

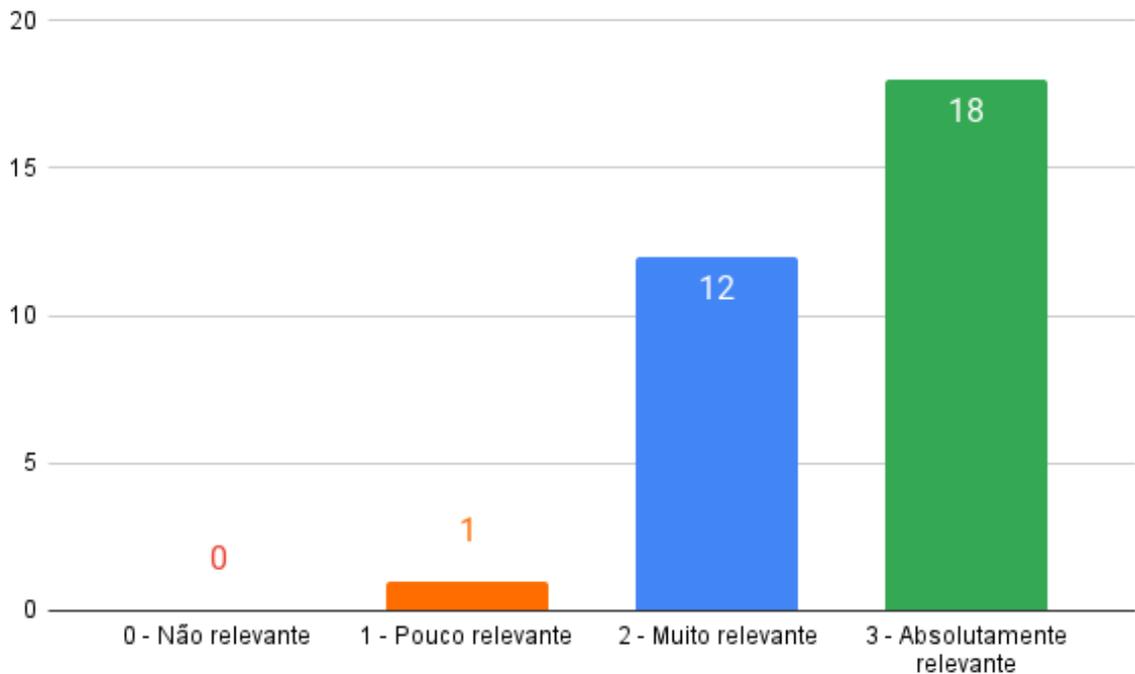


Figura 5.7 – Qual a relevância da prática 002 - Avaliar traços de personalidade e soft skills para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 84,95%, sendo essa a prática com a quarta maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

O comentário fornecido por um dos especialistas oferece uma perspectiva adicional:

A estabilidade do time depende de muitos traços e nuances que coletivamente são difíceis de prever a partir da personalidade de um membro do time. É muito difícil prever como todos os traços das personalidades do time se encaixam. (Especialista E27)

Enquanto testes e observações podem fornecer insights valiosos sobre um indivíduo, o comentário do especialista ressalta que é preciso reconhecer a complexidade das relações humanas e que a dinâmica do grupo podem ser influenciadas por uma combinação de fatores e não apenas pelas características individuais.

A Figura 5.8 mostra que na opinião dos especialistas, a prática de avaliar traços de

personalidade e soft skills é mais relevante para resolver questões relacionadas à satisfação dos indivíduos, embora nenhum comentário tenha sido adicionado relacionado a esse tema.

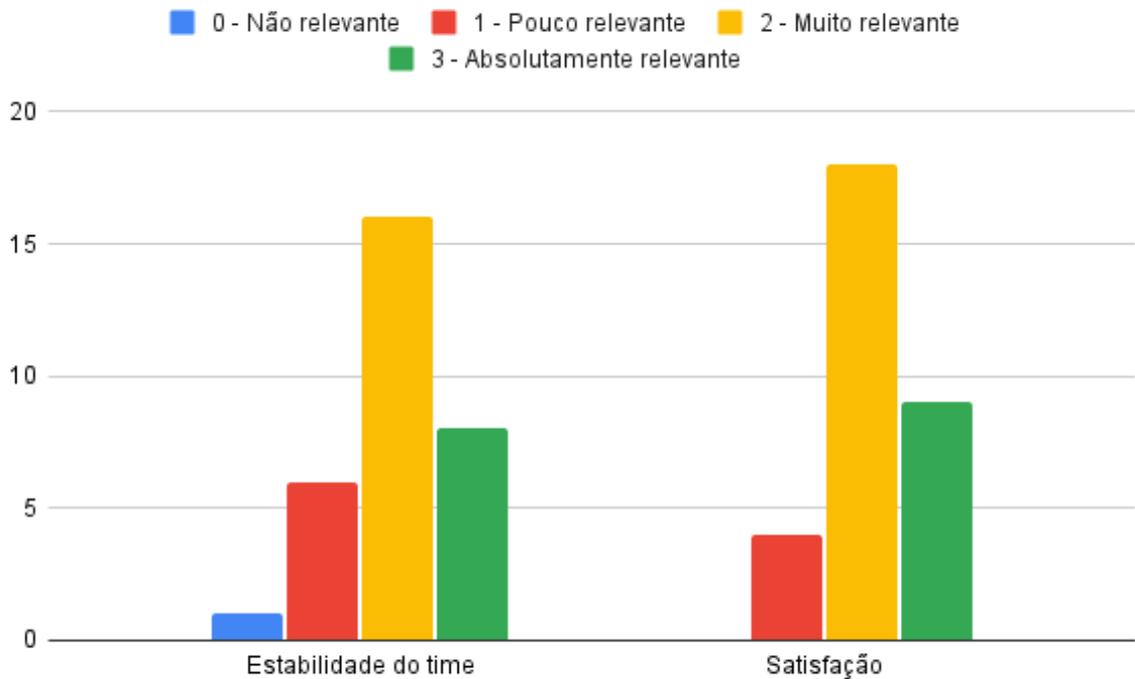


Figura 5.8 – Qual a relevância da prática 002 - Avaliar traços de personalidade e soft skills para resolver os problemas-alvo identificados?

5.2.3 Prática 003 - Executar checklist de onboarding

Sobre a relevância de executar checklist de onboarding no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.9:

- 0 especialistas consideram a prática como "não relevante";
- 2 especialistas (6,45%) consideram a prática como "pouco relevante";
- 13 especialistas (41,94%) consideram a prática como "muito relevante";
- 16 especialistas (51,61%) consideram a prática como "absolutamente relevante".

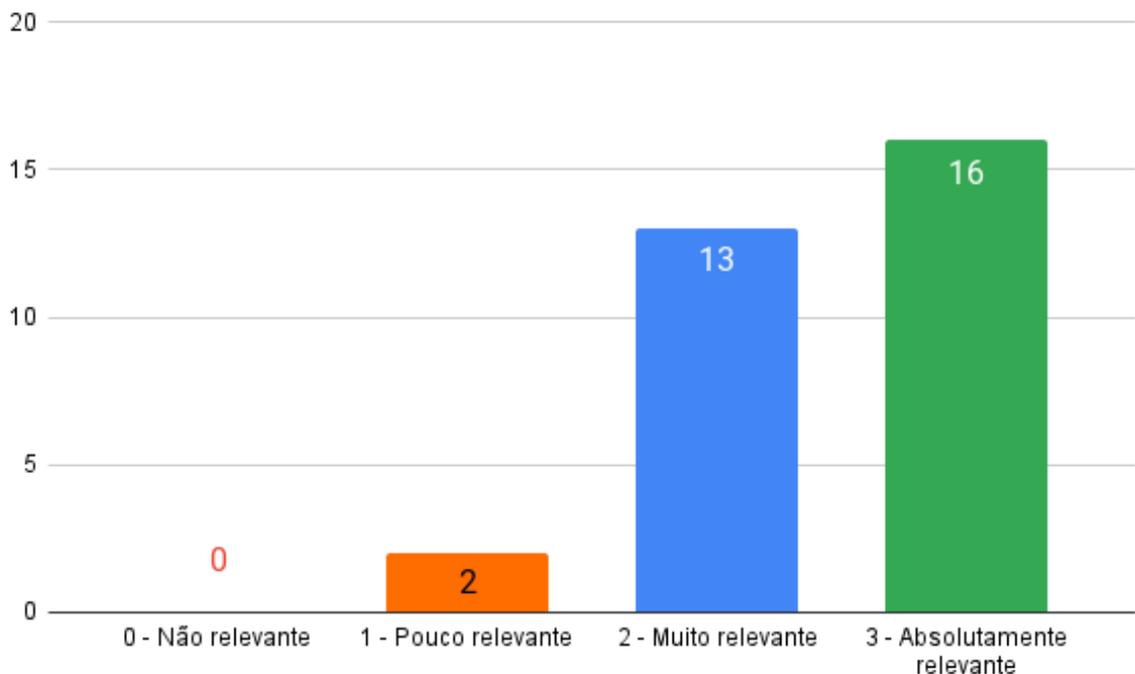


Figura 5.9 – Qual a relevância da prática 003 - Executar checklist de onboarding para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 81,72%, sendo essa a prática com a nona maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

Dois especialistas ofereceram comentários adicionais. O primeiro, entende que a prática é não relevante para o tema de habilidade e conhecimentos técnicos, argumentando que o onboarding estruturado não necessariamente influencia as habilidades ou o conhecimento técnico inerente do profissional.

Um processo de onboarding não necessariamente vai auxiliar no que o profissional tem como habilidade/conhecimento técnico. (Especialista E22)

O segundo especialista amplia a discussão para a autonomia, ressaltando que a habilidade de um indivíduo em encontrar soluções de forma independente é beneficiada pelo onboarding.

A autonomia beneficia-se tanto a nível organizacional quanto a nível de time, onde encontrar as respostas para suas dúvidas, por exemplo. Isso pode acarretar em um desempenho técnico melhor, mas não é uma relação direta, por isso vejo menos benefício na área de habilidades e conhecimentos técnicos. (Especialista E27)

Ambos especialistas argumentam que isso tem nenhuma ou pouca relevância, a evolução das habilidades e conhecimentos deve ser uma atividade contínua não limitada ao processo de

onboarding.

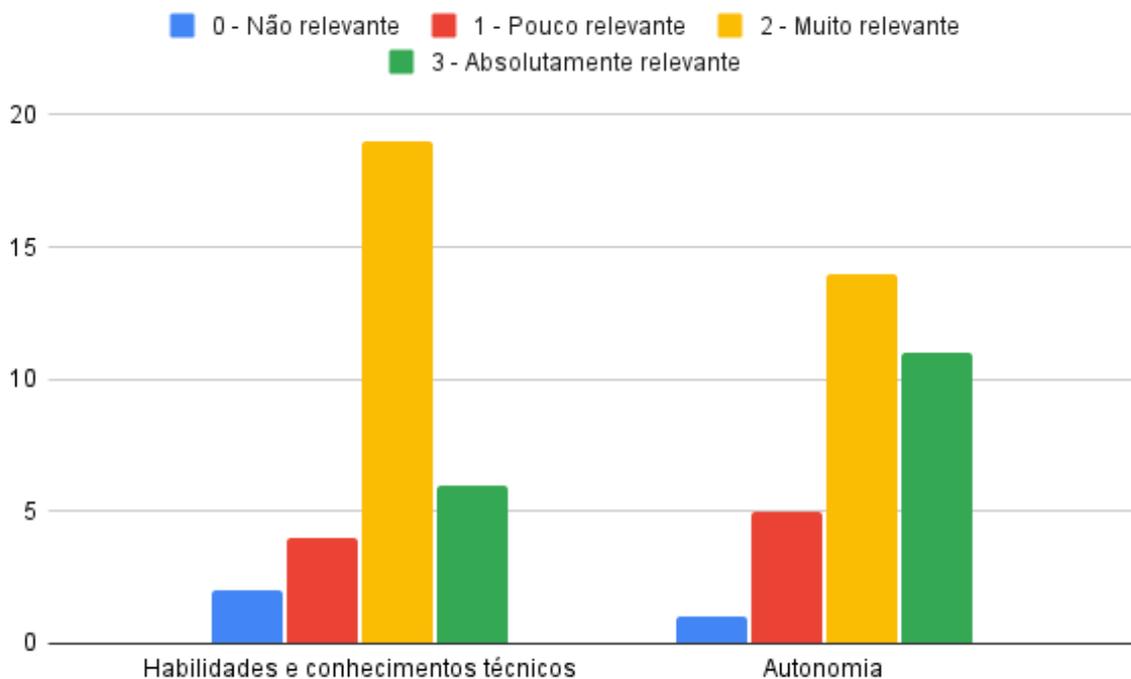


Figura 5.10 – Qual a relevância da prática 003 - Executar checklist de onboarding para resolver os problemas-alvo identificados?

As relevâncias calculadas foram de 64,52% para temas relacionados a habilidades e conhecimentos técnicos e 70,97% para temas relacionados à autonomia.

A prática do onboarding é valiosa para a integração organizacional, possuindo uma forte correlação com a capacidade do funcionário de operar de maneira independente e autônoma. Entretanto, a execução de um checklist de onboarding, tem baixa relevância para o desenvolvimento de habilidades técnicas, visto que essas competências são tipicamente adquiridas antes da chegada do profissional à organização ou serão desenvolvidas após a sua entrada.

Embora o percentual de relevância seja baixo, é notável que 19 especialistas, tenham considerado como muito relevante para o desenvolvimento de habilidades e conhecimento técnicos, não foram, porém, adicionados comentários que suportem esse entendimento, o autor supõe que o checklist de onboarding efetivo possa conter componentes de estudo e treinamento, que possibilitem um nivelamento inicial e preenchimento de lacunas de conhecimento, direcionando o desenvolvimento técnico contínuo do profissional.

Considerando os especialistas que consideraram a prática como não relevante, não foi possível identificar um perfil específico em termos de área de atuação, experiência profissional

ou formação acadêmica que possa sugerir um contexto específico onde estas práticas sejam consideradas menos relevantes.

5.2.4 Prática 004 - Criar canais e promover a comunicação aberta

Sobre a relevância de criar canais e promover a comunicação aberta no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.11:

- 0 especialistas consideram a prática como "não relevante";
- 0 especialistas consideram a prática como "pouco relevante";
- 9 (29,03%) consideram a prática como "muito relevante";
- 22 (70,97%) consideram a prática como "absolutamente relevante".

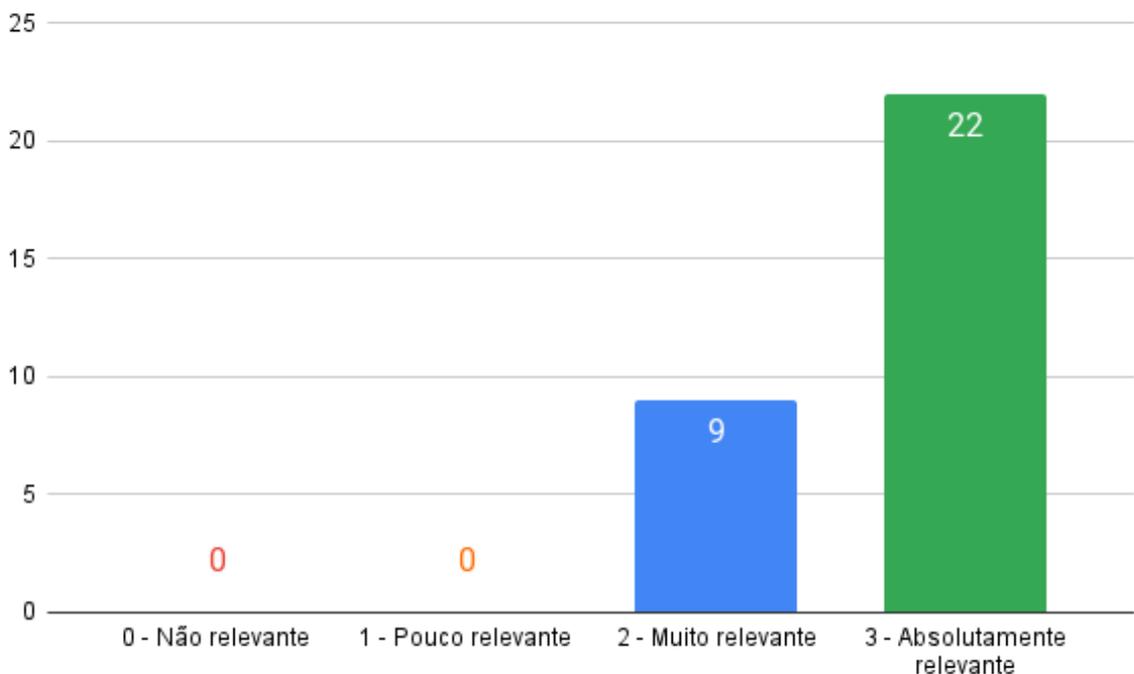


Figura 5.11 – Qual a relevância da prática 004 - Criar canais e promover a comunicação aberta para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 90,32%, sendo essa a prática com a segunda maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

Os comentários fornecidos por dois especialistas oferecem perspectivas adicionais:

Não ficou claro como aspectos geográficos são relevantes nesse contexto, talvez no sentido de times que não estão na mesma localização e precisem se comunicar de maneira mais eficiente e para isso canais de comunicação abertos e mais inclusivos possam auxiliar nesse quesito? (Especialista E3)

Absolutamente relevante, principalmente quando a atuação ocorre no modelo remoto, onde existe uma "barreira" na comunicação. (Especialista E26)

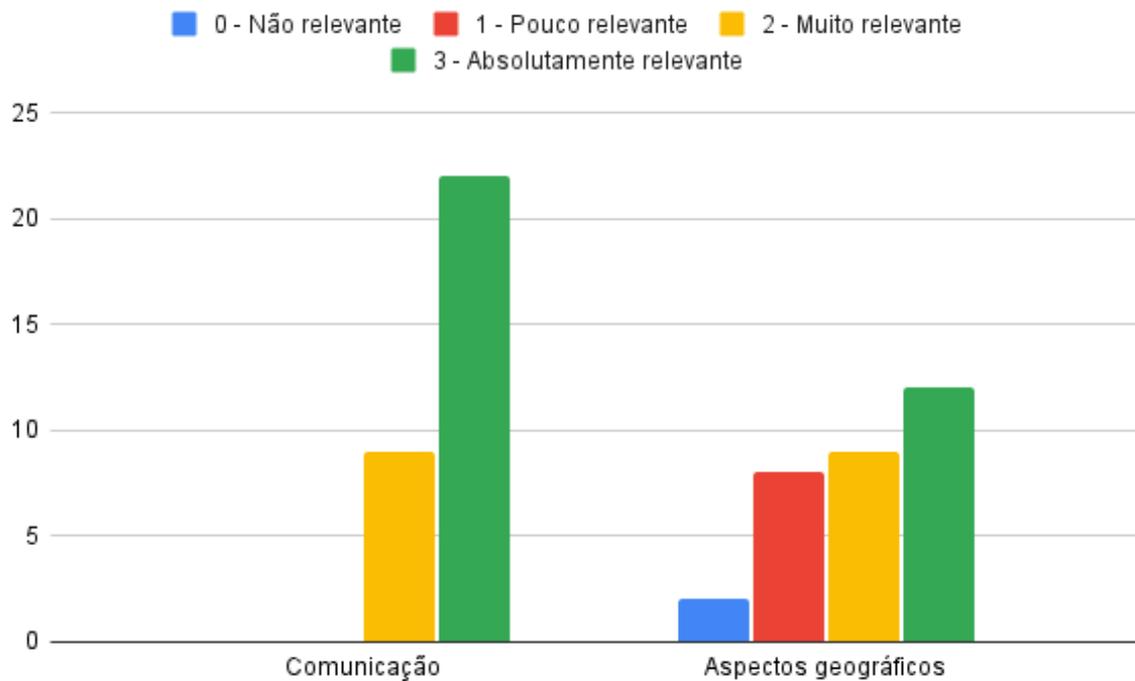


Figura 5.12 – Qual a relevância da prática 004 - Criar canais e promover a comunicação aberta para resolver os problemas-alvo identificados?

A prática de criar canais e promover a comunicação aberta é vista pelos especialistas como essencial, ela possui a maior relevância para os dois problemas alvo correlacionados previamente identificados. Sendo 90,32% relevante para os problema alvo relacionados a comunicação (o mesmo percentual de relevância da prática como um todo), e 66,67% para os problemas relacionados à aspectos geográficos.

Em relação aos Aspectos Geográficos, a relevância da prática é significativamente menor quando comparada às práticas mais relevantes para resolução de outros problemas-alvo. A literatura aponta que diferenças culturais, barreiras linguísticas e fusos horários são grandes obstáculos que as organizações enfrentam. A prática de comunicação aberta pode ajudar a superar essas barreiras, promovendo maior integração e colaboração entre equipes distribuídas. Contudo, fica claro que estratégias de comunicação devem ser complementadas com medidas

específicas que abordem a adequação cultural e treinamentos, para mitigar o distanciamento e maximizar a produtividade em cenários geograficamente distribuídos.

5.2.5 Prática 005 - Criar cronograma mínimo

Sobre a relevância de criar um cronograma mínimo no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.13:

- 0 especialistas consideram a prática como "não relevante";
- 0 especialistas consideram a prática como "pouco relevante";
- 14 especialistas (45,16%) consideram a prática como "muito relevante";
- 17 especialistas (54,84%) consideram a prática como "absolutamente relevante".

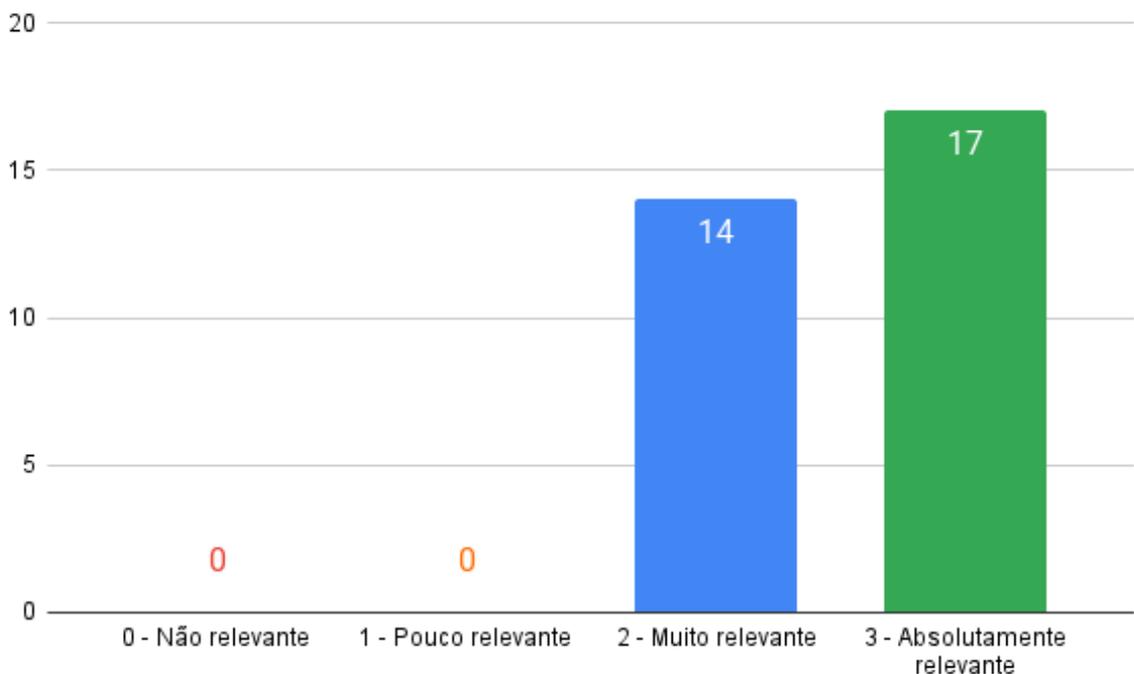


Figura 5.13 – Qual a relevância da prática 005 - Criar cronograma mínimo para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 84,95%, sendo essa a prática com a quinta maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

Os comentários fornecidos por dois especialistas oferecem perspectivas adicionais:

Controle do tempo. Muito relevante. (Especialista E11)

Esse ponto, na minha visão, ajuda a organizar minimamente o que vem pela frente para ser aplicado em nível de pessoas, imprescindível. (Especialista E27)

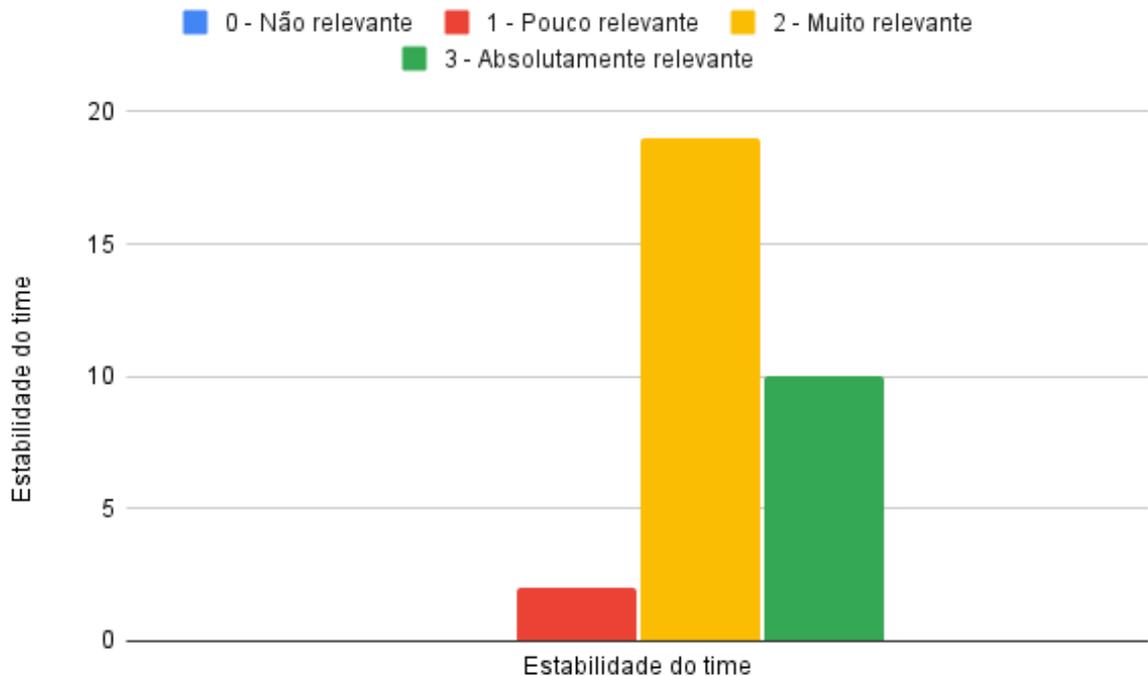


Figura 5.14 – Qual a relevância da prática 005 - Criar cronograma mínimo para resolver os problemas-alvo identificados?

Avaliando a relevância da criação de um cronograma mínimo no contexto específico da estabilidade do time, calculada em 75,27%, percebe-se uma lacuna em relação a sua relevância geral, calculada em 84,95%. Essa diferença torna-se evidente ao observar que um menor número de especialistas considera a prática como absolutamente relevante quando focada exclusivamente na estabilidade da equipe (17 de forma geral contra 10 para o problema alvo). Isso pode indicar a existência de outros problemas alvo que podem ser resolvidos através dessa prática, embora faltem comentários diretos que validem essa suposição.

5.2.6 Prática 006 - Criar job description para contratação

Sobre a relevância de criar a job description para contratação no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.15:

- 0 especialistas consideram a prática como "não relevante";
- 1 (3,23%) consideram a prática como "pouco relevante";
- 13 (41,93%) consideram a prática como "muito relevante";
- 17 especialistas (54,84%) consideram a prática como "absolutamente relevante".

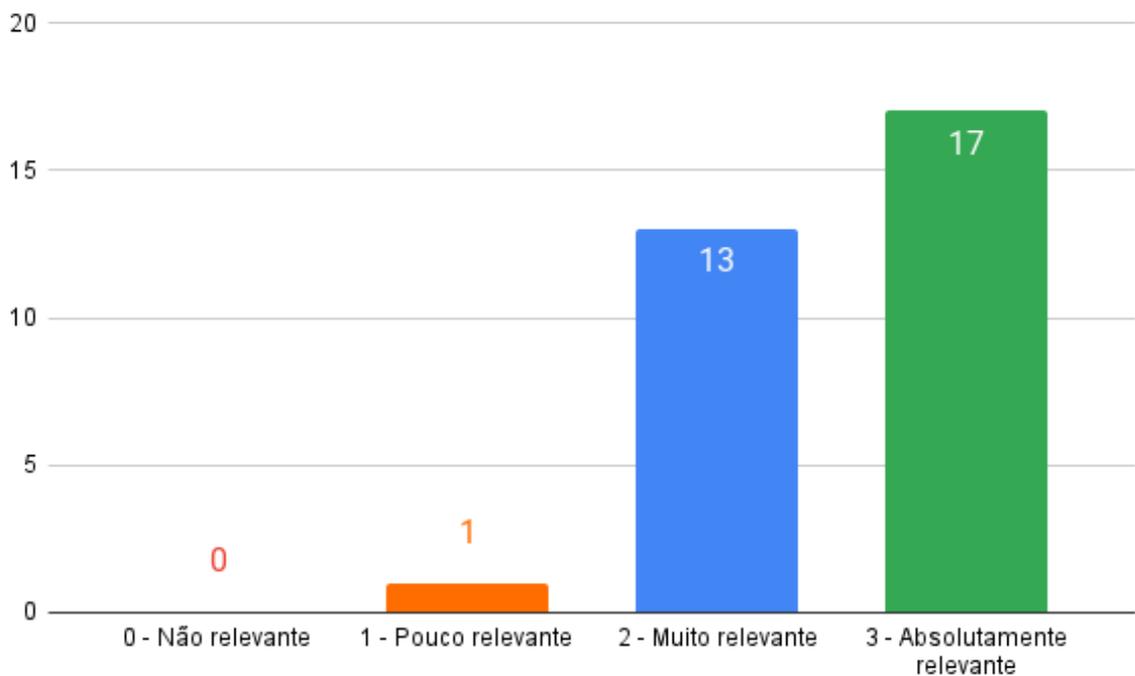


Figura 5.15 – Qual a relevância da prática 006 - Criar job description para contratação para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 83,87%, sendo essa a prática com a sexta maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

O comentário fornecido por um dos especialistas complementa que a relevância da job description no contexto de estabilidade do time, podendo ser utilizada como artefato para seleção de candidatos:

Não só o "job description" como também a aplicação de testes/validações técnicas. (Especialista E26)

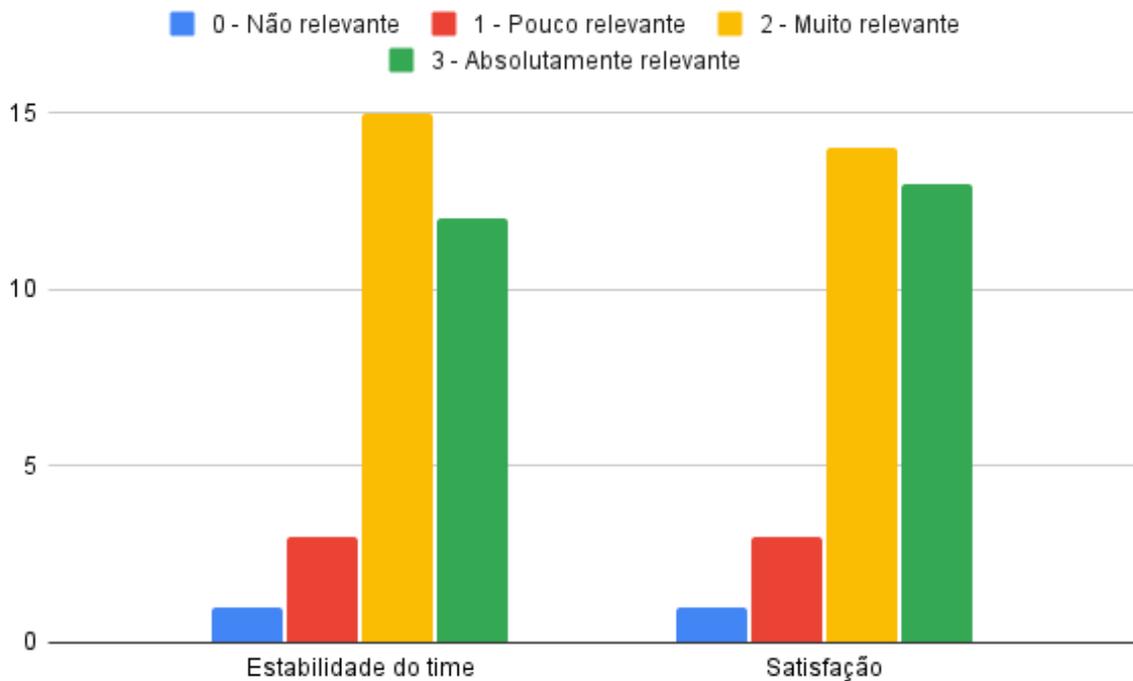


Figura 5.16 – Qual a relevância da prática 006 - Criar job description para contratação para resolver os problemas-alvo identificados?

A prática de criar job description claro, na opinião dos especialistas, apresenta relevâncias similares nos contextos de "Estabilidade do time"(74,19%) e "Satisfação"(75,27%). Essas porcentagens quase equivalentes indicam que a prática desempenha um papel quase igualmente significativo em abordar ambos os desafios.

Por um lado, descrições de trabalho detalhadas contribuem para a estabilidade da equipe ao facilitar a seleção de candidatos adequados e reduzir o turnover. Por outro lado, alinhar as expectativas dos funcionários com a realidade do trabalho, é crucial para a satisfação dos funcionários.

5.2.7 Prática 007 - Prover feedback ao time

Sobre a relevância de prover feedback ao time no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.17:

- 0 especialistas consideram a prática como "não relevante";
- 0 especialistas consideram a prática como "pouco relevante";

- 4 (12,90%) consideram a prática como "muito relevante";
- 27 (87,10%) consideram a prática como "absolutamente relevante".

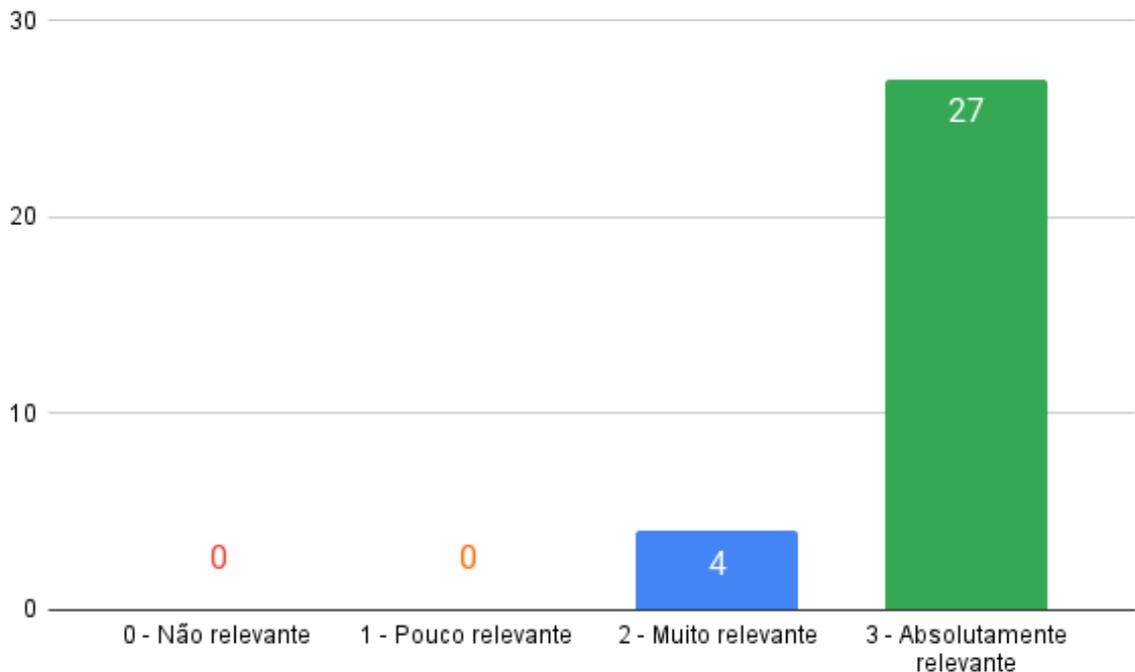


Figura 5.17 – Qual a relevância da prática 007 - Prover feedback ao time para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 95,70%, sendo essa a prática com a prática com maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

O feedback, conforme sugerido por um dos especialistas, pode ter um escopo de aplicação mais amplo do que considerado no momento da aplicação do questionário, de acordo com o seu comentário:

Feedback é uma ótima maneira de promover a satisfação da equipe. Eu colocaria e faria a proposta de incluir, não só "satisfação" e "comunicação" como problemas-alvo, mas "motivação" pois um feedback positivo e até mesmo um negativo com críticas construtiva podem motivar os colaboradores do time e mostra que o trabalho deles está sendo valorizado e que a gerência se importa com cada um. (Especialista E3)

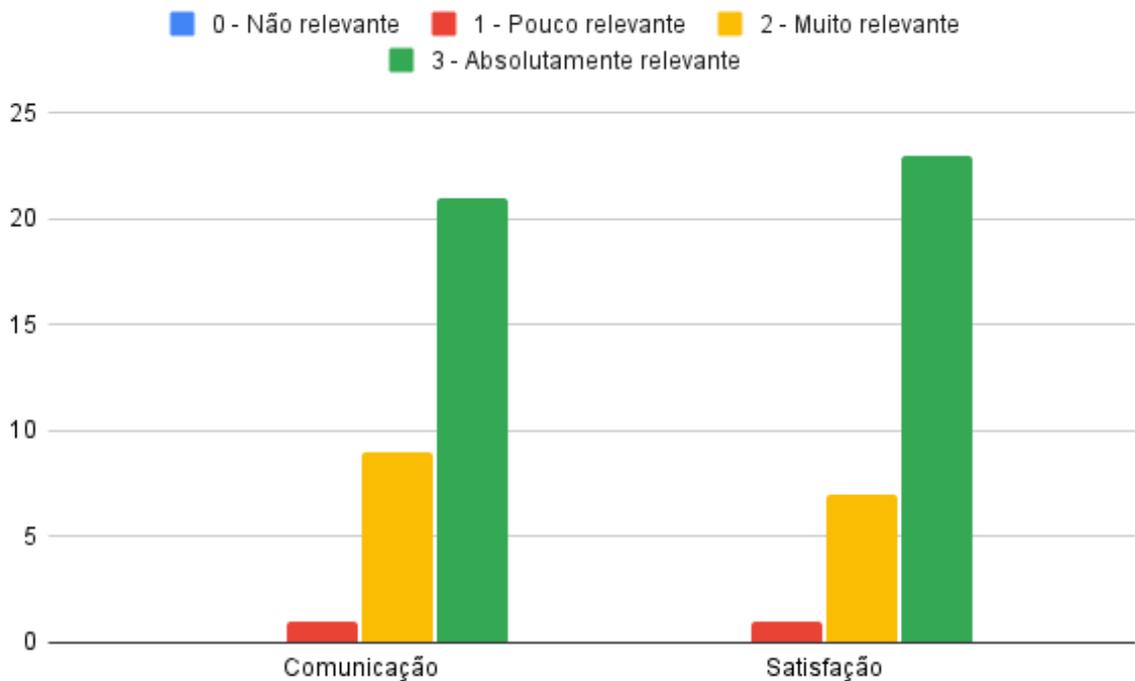


Figura 5.18 – Qual a relevância da prática 007 - Prover feedback ao time para contratação para resolver os problemas-alvo identificados?

A prática de prover feedback ao time, na opinião dos especialistas, apresenta altas relevâncias nos contextos de "Comunicação"(88,17%) e "Satisfação"(90,32%), sendo esta a segunda com maior relevância para problemas de comunicação e a principal para problemas relacionados à satisfação, sendo percebida como geradora de crescimento pessoal e profissional, reconhecimento, e melhorando as relações com colegas e superiores.

5.2.8 Prática 008 - Definir tamanho do time

Sobre a relevância de definir tamanho do time no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.19:

- 0 especialistas consideram a prática como "não relevante";
- 2 (6,45%) consideram a prática como "pouco relevante";
- 11 (35,48%) consideram a prática como "muito relevante";
- 18 (58,07%) consideram a prática como "absolutamente relevante".

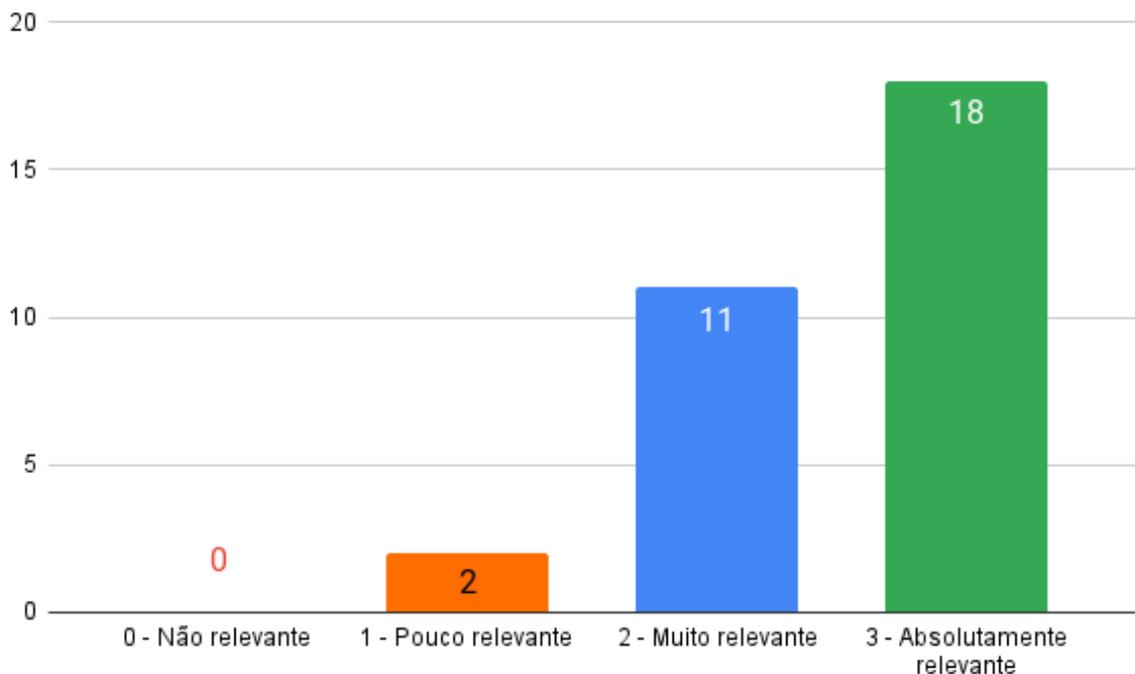


Figura 5.19 – Qual a relevância da prática 008 - Definir tamanho do time para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 83,87%, sendo essa a prática com a sétima maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

Nenhum dos 31 especialistas respondentes do questionário adicionaram comentários adicionais para a prática.

Embora Stylianou e Andreou (2016) apontem uma relação inversamente proporcional entre o tamanho do time e a eficácia da comunicação, os especialistas que responderam ao questionário indicam que, na prática, este fator não é tão determinante. A prática de definir o tamanho do time, com uma relevância calculada em 67,74% no contexto da comunicação, foi considerada a de menor relevância dentre as práticas destinadas a resolver problemas de comunicação em projetos de desenvolvimento de software. Isso sugere que, na percepção dos especialistas, outros fatores podem ter um impacto mais significativo na comunicação do que o número de membros na equipe.

Quanto à Satisfação, com relevância calculada em 77,42%, uma equipe dimensionada adequadamente pode contribuir para um melhor equilíbrio entre carga de trabalho e recursos disponíveis, impactando na satisfação dos membros da equipe. Quando o tamanho do time é

otimizado, os membros tendem a sentir-se mais valorizados, reconhecidos e capazes de alcançar crescimento pessoal e profissional.

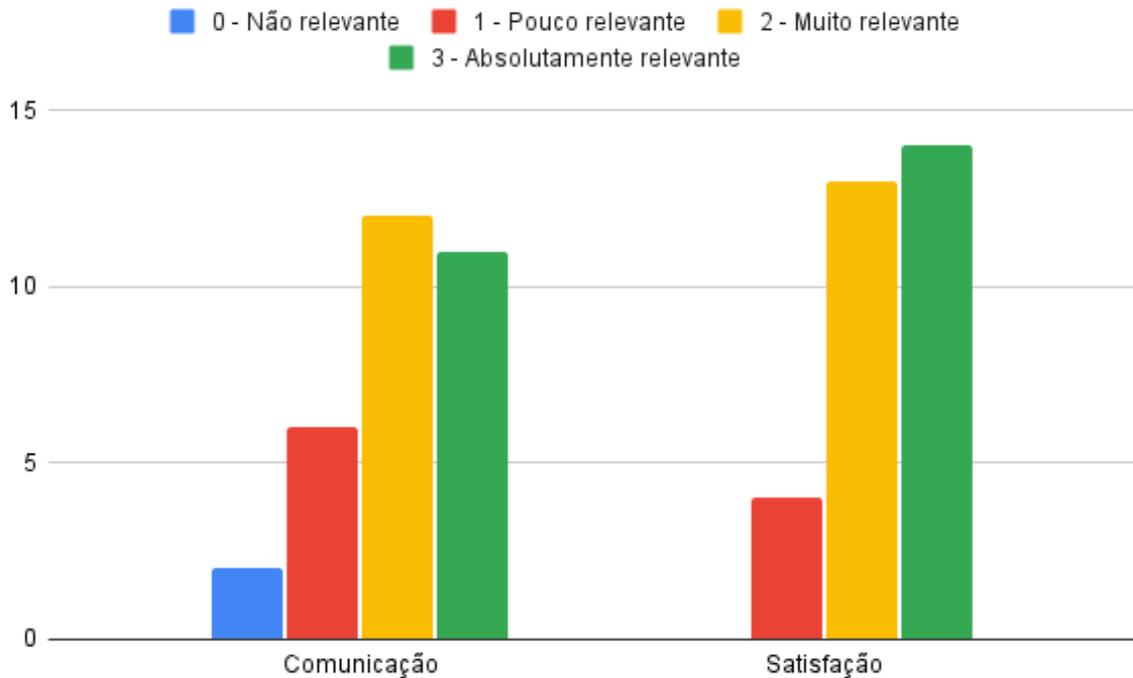


Figura 5.20 – Qual a relevância da prática 008 - Definir tamanho do time para contratação para resolver os problemas-alvo identificados?

5.2.9 Prática 009 - Desafiar time a aprender novas habilidades

Sobre a relevância de desafiar time a aprender novas habilidades no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.21:

- 0 especialistas consideram a prática como "não relevante";
- 1 (3,23%) consideram a prática como "pouco relevante";
- 18 (58,07%) consideram a prática como "muito relevante";
- 12 (38,70%) consideram a prática como "absolutamente relevante".

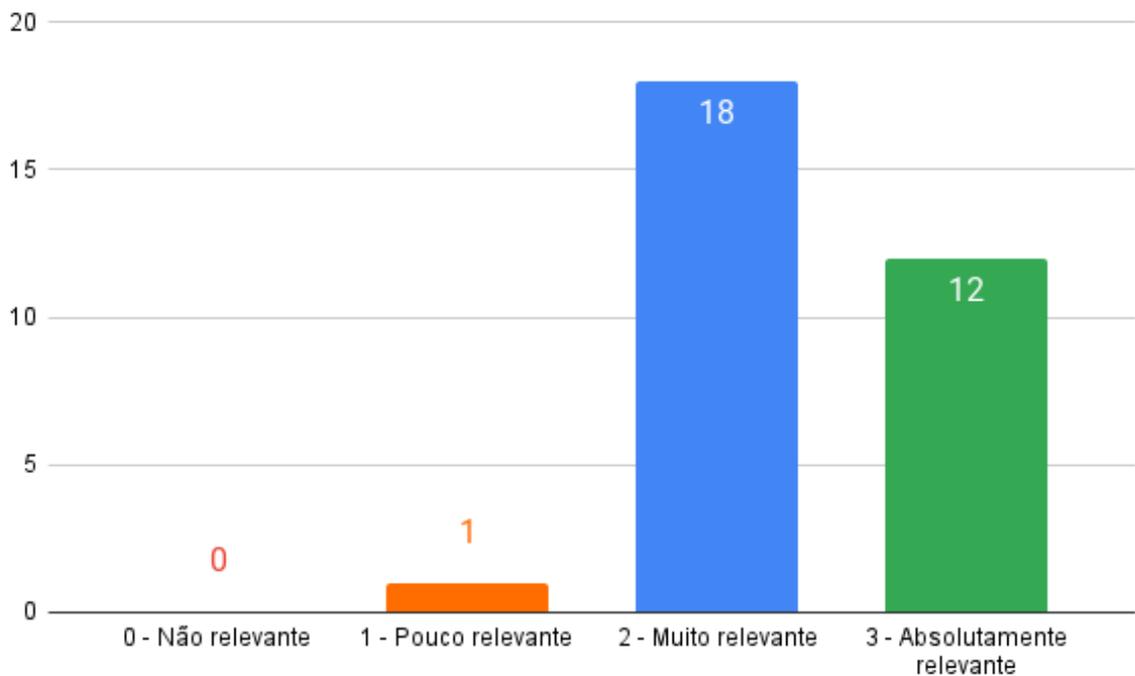


Figura 5.21 – Qual a relevância da prática 009 - Desafiar time a aprender novas habilidades para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 78,49%, sendo essa a prática com a décima primeira maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

Nenhum dos 31 especialistas respondentes do questionário adicionaram comentários adicionais para a prática.

A relevância da prática no contexto de problemas relacionados à satisfação das pessoas foi calculada em 78,49%, correspondendo exatamente à relevância geral da prática. Embora esta relevância no contexto da satisfação seja considerável, ela acaba sendo ofuscada pela relevância da prática no contexto dos outros problemas-alvo: motivação (83,87%) e habilidades e conhecimentos técnicos (87,10%), superam a relevância geral da prática. Esta diferença ressalta a importância ainda maior da implementação da prática particularmente nesses dois últimos contextos.

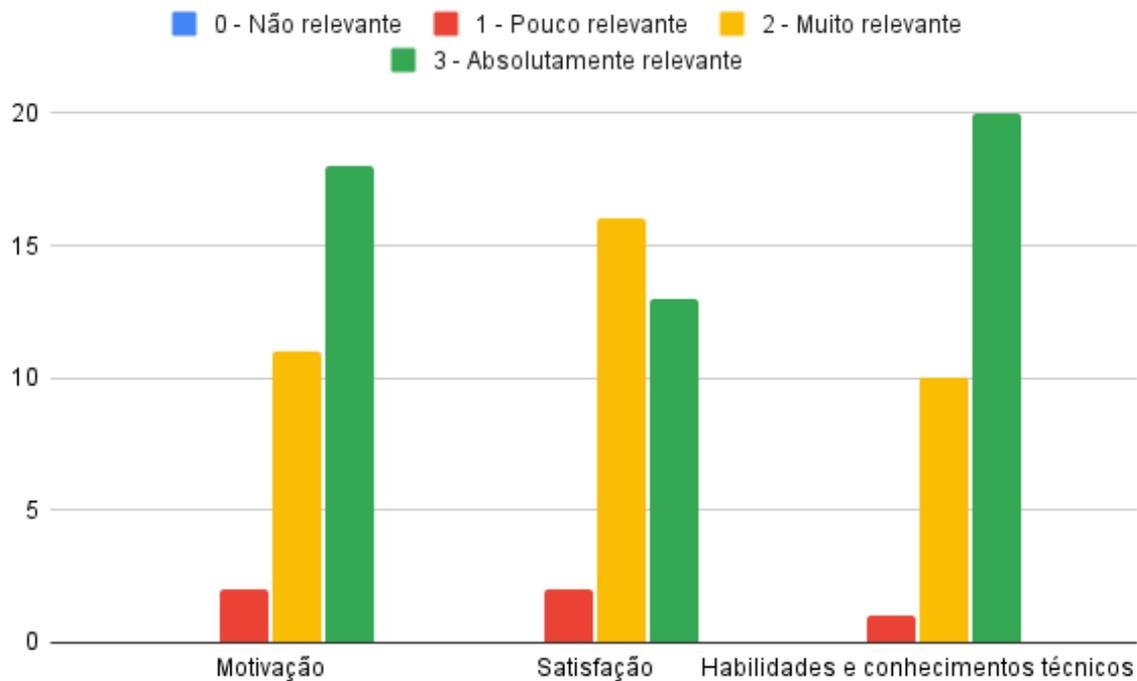


Figura 5.22 – Qual a relevância da prática 009 - Desafiar time a aprender novas habilidades para contratação para resolver os problemas-alvo identificados?

A prática de desafiar time a aprender novas habilidades, foi considerada pelos especialistas, como a mais relevante no contexto de habilidades e conhecimentos técnicos.

5.2.10 Prática 010 - Praticar gestão à vista

Sobre a relevância de praticar gestão à vista no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.23:

- 0 especialistas consideram a prática como "não relevante";
- 2 (6,45%) consideram a prática como "pouco relevante";
- 9 (29,03%) consideram a prática como "muito relevante";
- 20 (64,52%) consideram a prática como "absolutamente relevante".

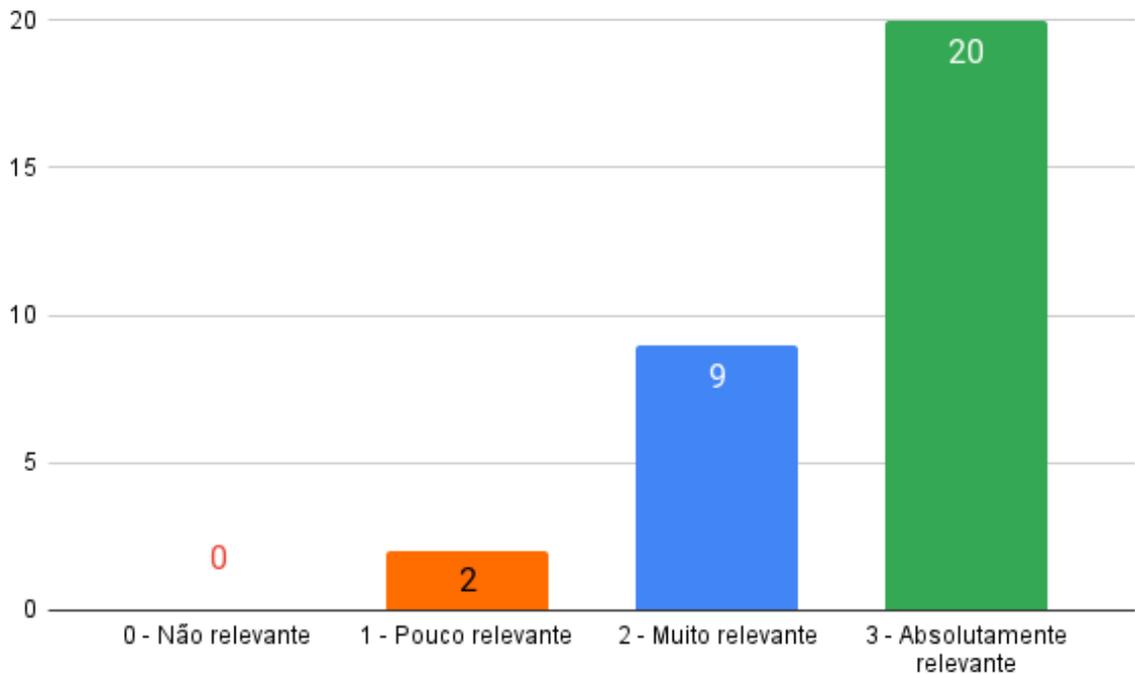


Figura 5.23 – Qual a relevância da prática 010 - Praticar gestão à vista para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 86,02%, sendo essa a prática com a terceira maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

O comentário fornecido por um dos especialistas oferece perspectivas adicionais:

A gestão visual deve ser medida de maneira precisa para assegurar a exatidão das informações utilizadas para embasar decisões e orientar a equipe. colocar um excessivo foco em métricas superficiais pode resultar na desmotivação da equipe e direcionar a atenção para problemas imediatos, em vez de abordar os desafios de maneira estratégica. (Especialista E21)

O comentário do especialista ressalta a necessidade de planejamento para o emprego da prática, alertando sobre o risco de focar em métricas que podem desviar a atenção dos desafios estratégicos. Embora a prática seja de grande importância, com impactos em diversos tipos de problema, sua eficácia depende da qualidade e relevância das informações expostas.

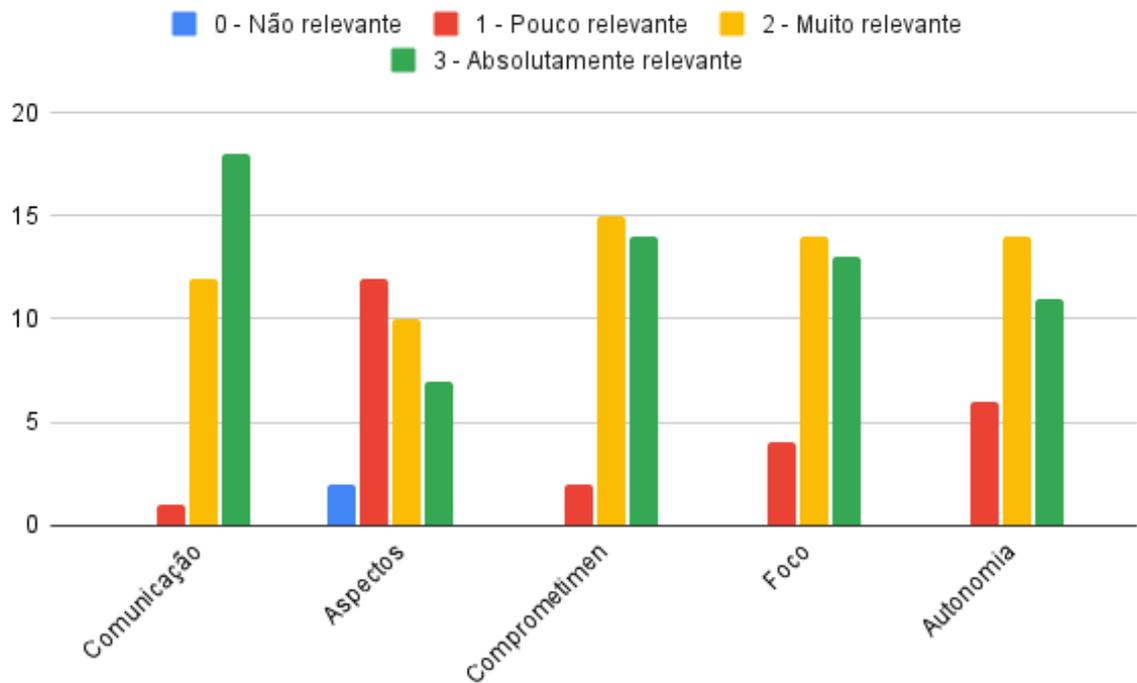


Figura 5.24 – Qual a relevância da prática 010 - Praticar gestão à vista para resolver os problemas-alvo identificados?

Praticar gestão à vista é altamente relevante para melhorar a comunicação, relevância calculada em 84,95%. Em projetos de desenvolvimento de software, onde a comunicação eficaz é crítica, seu emprego pode facilitar o fluxo de informações e minimizar os desafios associados ao tamanho do time.

No contexto de problemas associados a aspectos geográficos, a relevância, na opinião dos especialistas, foi considerada baixa, 56,99%. Embora auxilie a resolver problemas de comunicação gerados pela distribuição geográfica do time, por exemplo, outros desafios como diferenças culturais e barreiras linguísticas, não são beneficiados.

O comprometimento do time é significativamente impactado pela gestão à vista, relevância de 79,57%, sugerindo que a visualização clara dos objetivos e progresso do time pode aumentar a responsabilidade dos indivíduos e engajamento da equipe. Embora numericamente a relevância seja inferior a de outros problemas-alvo, a prática de gestão à vista foi considerada pelos especialistas como a principal para enfrentar problemas de comprometimento.

A relevância da prática no contexto de foco foi calculada em 76,32%, devendo-se principalmente a ofertar uma forma de manter o foco da equipe alinhado e concentrado nas metas atuais.

Por fim, no contexto de autonomia, calculada em 72,04% de relevância. A prática pode permitir que os membros da equipe tomem decisões informadas e se auto-organizem de forma eficaz, fortalecendo a sensação de controle sobre seu trabalho.

5.2.11 Prática 011 - Identificar e gerenciar competências do time

Sobre a relevância de identificar e gerenciar competências do time no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.25:

- 0 especialistas consideram a prática como "não relevante";
- 0 especialistas consideram a prática como "pouco relevante";
- 16 especialistas (51,61%) consideram a prática como "muito relevante";
- 15 especialistas (48,39%) consideram a prática como "absolutamente relevante".

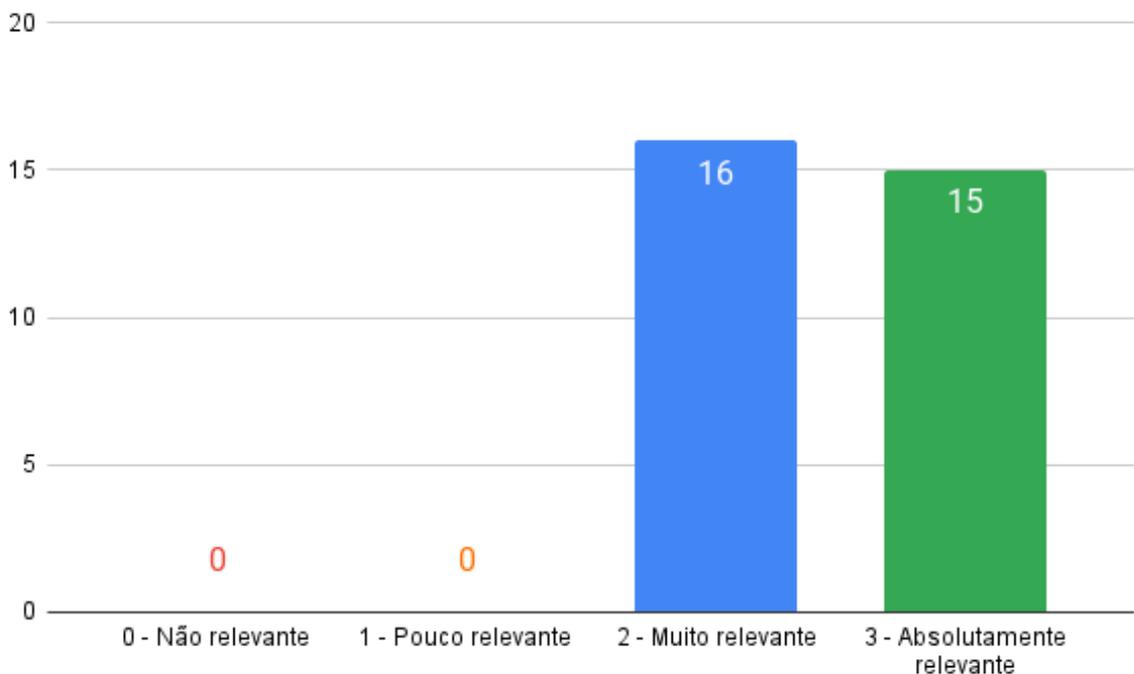


Figura 5.25 – Qual a relevância da prática 011 - Identificar e gerenciar competências do time para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 78,49%, sendo essa a prática com a décima segunda maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

Nenhum dos 31 especialistas respondentes do questionário adicionaram comentários adicionais para a prática.

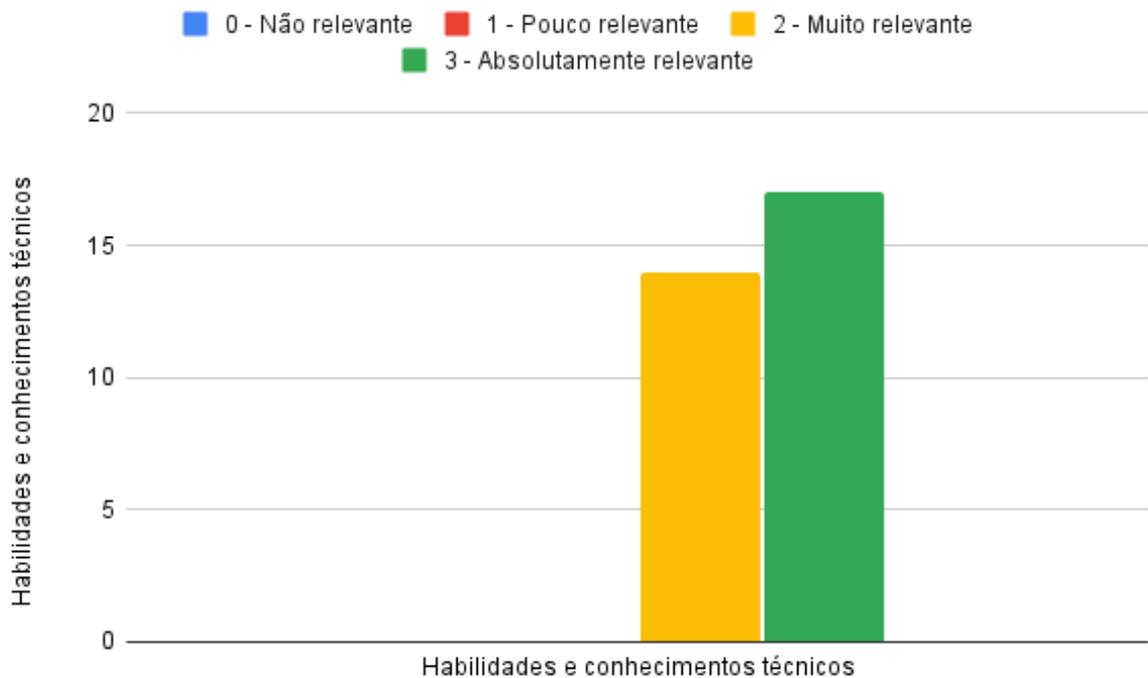


Figura 5.26 – Qual a relevância da prática 011 - Identificar e gerenciar competências do time para resolver os problemas-alvo identificados?

A relevância do emprego da prática de identificar e gerenciar competências do time no contexto de habilidades e conhecimentos técnicos, foi calculada em 84,95%, superando a relevância geral da prática em mais de seis pontos percentuais. Isso indica, que na opinião dos especialistas, há uma grande eficácia da prática em resolver problemas técnicos específicos, mesmo que ela não seja considerada tão importante quanto outras práticas de gestão de pessoas em projetos de desenvolvimento de software.

Não foram encontradas outros contextos correlacionados onde essa prática seria efetiva.

5.2.12 Prática 012 - Montar plano de carreira e plano de sucessão

Sobre a relevância de montar plano de carreira e plano de sucessão no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.27:

- 0 especialistas consideram a prática como "não relevante";
- 2 (6,45%) consideram a prática como "pouco relevante";
- 16 (51,61%) consideram a prática como "muito relevante";
- 13 especialistas (41,94%) consideram a prática como "absolutamente relevante".

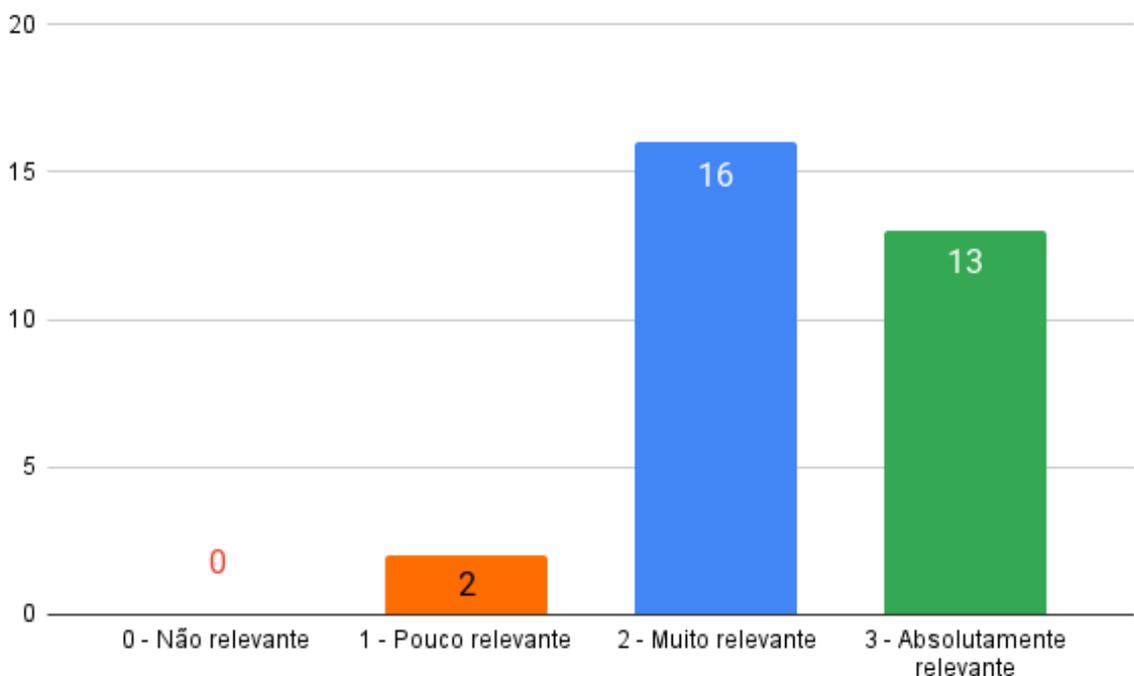


Figura 5.27 – Qual a relevância da prática 012 - Montar plano de carreira e plano de sucessão para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 78,49%, sendo essa a prática com a décima terceira maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

Nenhum dos 31 especialistas respondentes do questionário adicionaram comentários adicionais para a prática.

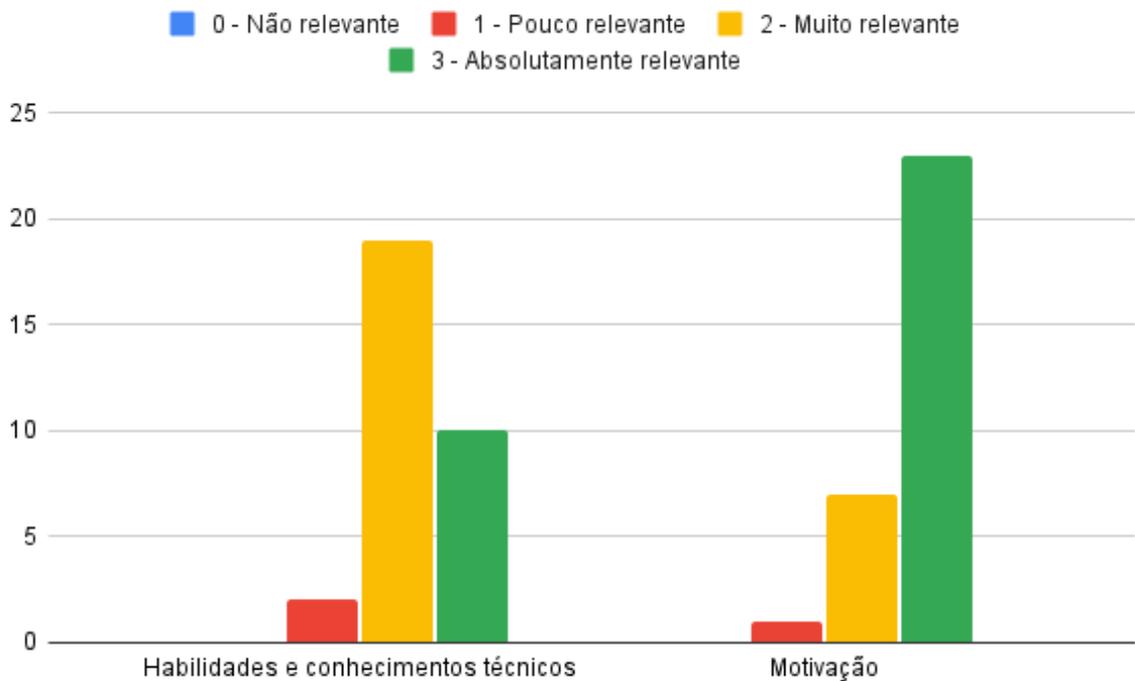


Figura 5.28 – Qual a relevância da prática 012 - Montar plano de carreira e plano de sucessão para resolver os problemas-alvo identificados?

Ao analisar a prática de "Montar plano de carreira e plano de sucessão" no contexto dos problemas alvo específicos, habilidades e conhecimento técnico (75,27%) e motivação (90,32%), percebemos que a relevância para o problema de "Habilidades e conhecimento técnico" possui uma distribuição de respostas similar às respostas sobre a relevância geral da prática.

Por outro lado, a relevância para "Motivação" é notavelmente mais alta do que a relevância geral, sendo essa a principal prática para tratamento de questões relacionadas à motivação do time. Planos de carreira claros e a possibilidade de progressão podem atender a várias motivações intrínsecas e extrínsecas, como o desejo de desenvolvimento profissional, reconhecimento e segurança no trabalho.

5.2.13 Prática 013 - Montar time heterogêneo

Sobre a relevância de montar time heterogêneo no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.29:

- 3 especialistas (9,68%) consideram a prática como "não relevante";
- 1 especialistas (3,23%) consideram a prática como "pouco relevante";
- 13 especialistas (41,93%) consideram a prática como "muito relevante";
- 14 especialistas (45,16%) consideram a prática como "absolutamente relevante".

A prática "Montar time heterogêneo" se destaca por ter recebido a maior quantidade de avaliações como "não relevante" dentre as dezesseis práticas analisadas, segundo a percepção dos especialistas.

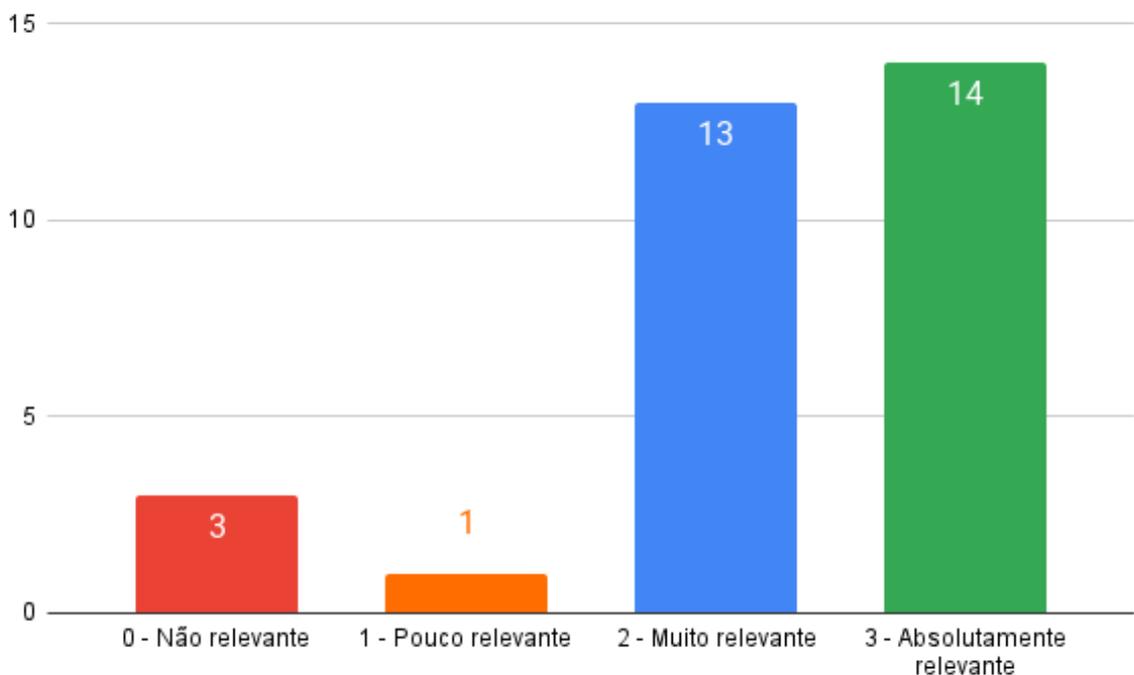


Figura 5.29 – Qual a relevância da prática 013 - Montar time heterogêneo para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 74,19%, sendo essa a prática rankeada número 15 entre as 16 práticas identificadas (Tabela 5.3).

Essa classificação sugere que, embora reconhecida pela literatura como importante, ela pode não ser vista como prioritária ou tão crítica quanto outras práticas nos projetos de desenvolvimento de software.

A falta de comentários adicionais deixa espaço para discussões sobre a percepção de relevância por parte dos especialistas, considerando a notável divergência nas respostas onde alguns especialistas categorizaram a prática como não relevante, que sugere que a eficácia e o valor da diversidade em equipes de desenvolvimento de software não são vistos de maneira uniforme. Enquanto alguns podem reconhecer os benefícios da diversidade para inovação e resolução criativa de problemas, outros parecem questionar seu impacto direto e imediato na eficiência dos projetos de desenvolvimento de software.

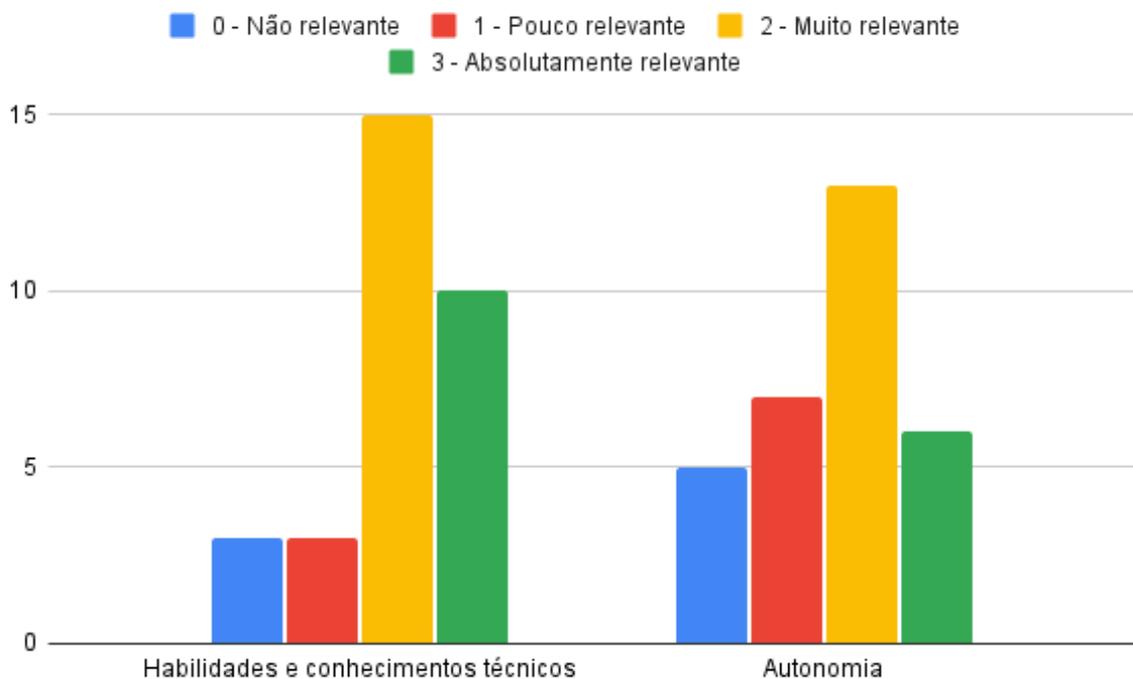


Figura 5.30 – Qual a relevância da prática 013 - Montar time heterogêneo para resolver os problemas-alvo identificados?

Comparando a relevância geral com as avaliações de relevância para os problemas alvo, observa-se que a prática é mais valorizada no contexto de "Habilidades e conhecimentos técnicos"(67,74%) do que em "Autonomia"(54,84%). Esta diferença pode indicar uma percepção de que, enquanto a diversidade é benéfica para ampliar a gama de habilidades e conhecimento disponíveis na equipe, ela pode não ser tão influente na promoção da autonomia dos membros da equipe.

5.2.14 Prática 014 - Organizar treinamentos

Sobre a relevância de organizar treinamentos no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.31:

- 0 especialistas consideram a prática como "não relevante";
- 1 especialistas (3,23%) consideram a prática como "pouco relevante";
- 17 especialistas (54,84%) consideram a prática como "muito relevante";
- 13 especialistas (41,93%) consideram a prática como "absolutamente relevante".

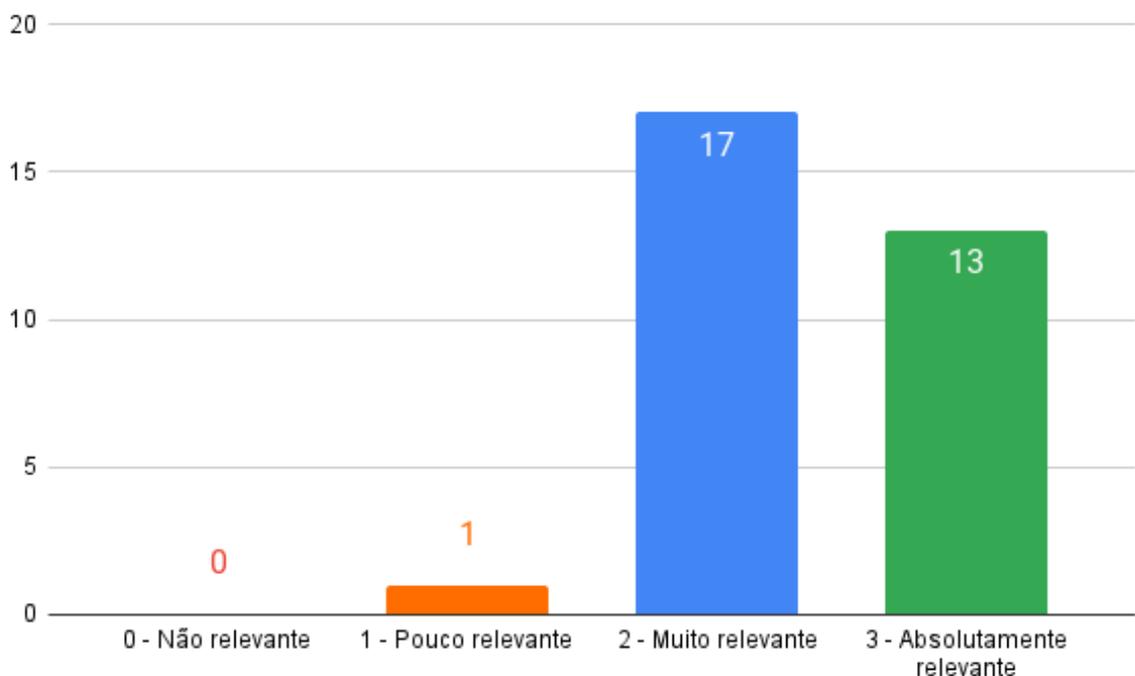


Figura 5.31 – Qual a relevância da prática 014 - Organizar treinamentos para gestão de pessoas em projetos de desenvolvimento de software?

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 79,57%, sendo essa a prática com a décima maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

Nenhum dos 31 especialistas respondentes do questionário adicionaram comentários adicionais para a prática.

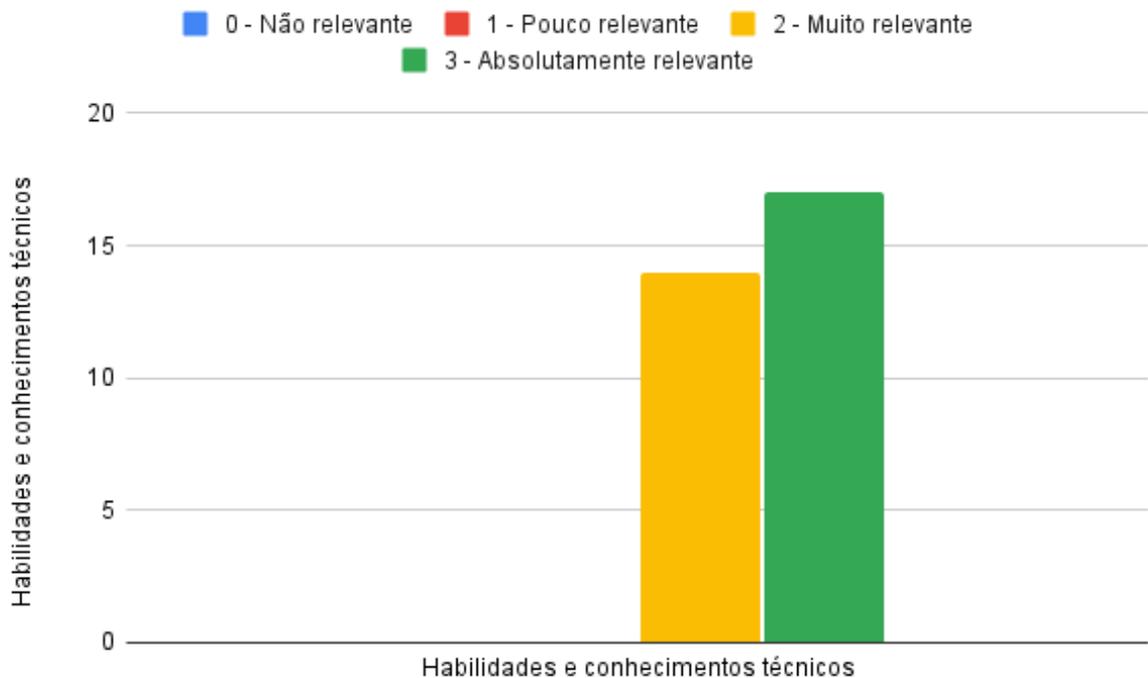


Figura 5.32 – Qual a relevância da prática 014 - Organizar treinamentos para resolver os problemas-alvo identificados?

No contexto do problema alvo de "Habilidades e conhecimento técnico" nos projetos de desenvolvimento de software, a prática "Organizar treinamentos" foi avaliada pelos especialistas com uma relevância de 84,95%, um valor mais alto do que a relevância geral da prática, calculada em 79,57%. Essa diferença sinaliza um reconhecimento da importância crítica dos treinamentos na melhoria das habilidades técnicas e do conhecimento dos profissionais em projetos de software. No entanto, apesar dessa alta relevância no contexto específico, a prática não foi identificada como a principal solução para os problemas relacionados a habilidades e conhecimentos técnicos. Isso sugere que, embora considerada importante, os especialistas podem não priorizar a organização de treinamentos tanto no contexto geral quanto no específico, indicando a existência de outras práticas percebidas como mais eficazes ou críticas para abordar esses desafios específicos.

No contexto específico de habilidades e conhecimentos técnicos, a prática de organizar treinamentos é percebida como menos relevante do que desafiar o time a aprender novas habilidades (ver Seção 5.2.9). É possível argumentar que a principal diferença está em quem assume a responsabilidade pelo desenvolvimento técnico. Enquanto a organização de treinamentos enfatiza no papel de uma entidade externa em prover o treinamento, a prática vista como mais

relevante pelos especialistas mantém o protagonismo no próprio indivíduo. Essa comparação indica uma preferência por estratégias de aprendizado autodirigidas, contrastando com métodos tradicionais baseados em treinamentos formais.

5.2.15 Prática 015 - Rotacionar time

Sobre a relevância de rotacionar o time no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.33:

- 1 especialistas (3,23%) consideram a prática como "não relevante";
- 5 especialistas (16,13%) consideram a prática como "pouco relevante";
- 16 especialistas (51,61%) consideram a prática como "muito relevante";
- 9 especialistas (29,03%) consideram a prática como "absolutamente relevante".

A prática "Rotacionar Time" se destaca por ter recebido a maior quantidade de avaliações como "pouco relevante" dentre as dezesseis práticas analisadas, segundo a percepção dos especialistas.

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 68,82%, sendo essa a prática com a menor relevância dentre as 16 práticas identificadas (Tabela 5.3).

A prática mostra maior relevância no desenvolvimento de "Habilidades e Conhecimentos Técnicos" (80,65%), mas parece ter um impacto menor na "Estabilidade do Time" (51,61%). Essa observação ressalta as preocupações sobre como a rotação frequente pode afetar negativamente a coesão e a eficiência da equipe, particularmente no que diz respeito a questões de relacionamento interpessoal e desafios técnicos, algo já identificado pela literatura, principalmente no que tange a rotação *project-to-project*, onde os indivíduos mudam de time.

Por outro lado, a prática apresenta uma relevância moderada nos problemas relacionados a "Autonomia" (65,59%), sugerindo que, enquanto pode promover a autonomia, dando aos membros da equipe maior liberdade e capacidade de se auto-organizarem, também pode ameaçar a estabilidade necessária para a manutenção de um trabalho coeso e bem coordenado.

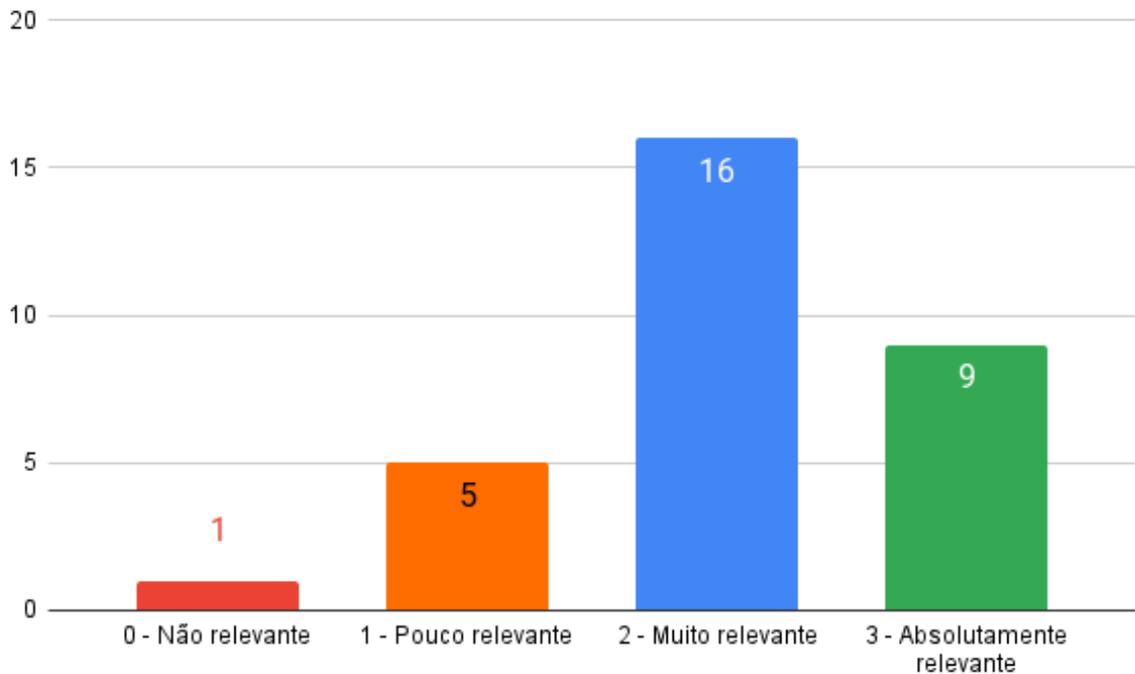


Figura 5.33 – Qual a relevância da prática 015 - Rotacionar time para gestão de pessoas em projetos de desenvolvimento de software?

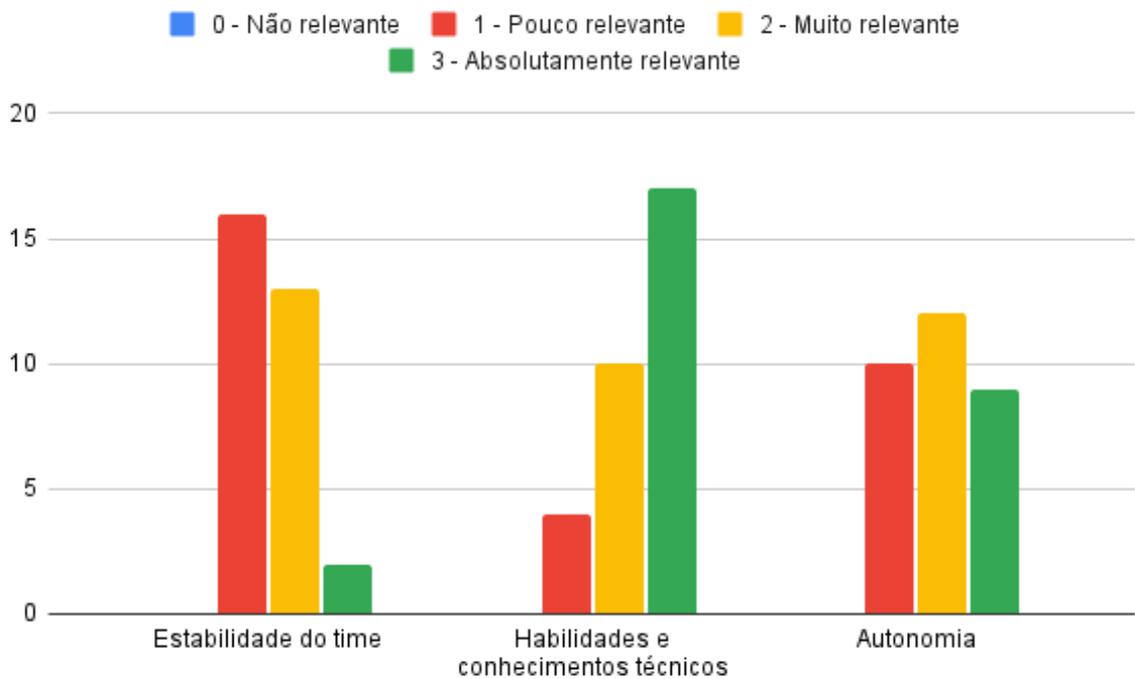


Figura 5.34 – Qual a relevância da prática 015 - Rotacionar time para resolver os problemas-alvo identificados?

O comentário fornecido por um dos especialistas oferece perspectivas adicionais:

Estabilidade do time marcada como "pouco relevante" pois em algumas experiências já conversei com times que tiveram problemas devido diversas rotação. Problemas de relacionamento e técnico. (Especialista E26)

Portanto, embora a rotação de time possa ser benéfica para o aprimoramento de habilidades técnicas, é essencial sua aplicação cuidadosa, atendendo às necessidades de estabilidade e autonomia da equipe, para aproveitar ao máximo seus benefícios e reduzir riscos potenciais.

Considerando os especialistas que consideraram a prática como não relevante, não foi possível identificar um perfil específico em termos de área de atuação, experiência profissional ou formação acadêmica que possa sugerir um contexto específico onde estas práticas sejam consideradas menos relevantes.

5.2.16 Prática 016 - Usar indicadores para monitorar a performance do time de desenvolvimento de software

Sobre a relevância de usar indicadores para monitorar a performance do time no contexto geral de gestão de pessoas em projetos de desenvolvimento de software, os especialistas responderam conforme descrito abaixo e demonstrado na Figura 5.35:

- 0 especialistas consideram a prática como "não relevante";
- 2 especialistas (6,45%) consideram a prática como "pouco relevante";
- 17 especialistas (54,84%) consideram a prática como "muito relevante";
- 12 especialistas (38,71%) consideram a prática como "absolutamente relevante".

Usando os pesos atribuídos a cada resposta, foi calculado um índice de relevância geral de 77,42%, sendo essa a prática com a décima quarta maior relevância dentre as 16 práticas identificadas (Tabela 5.3).

As relevâncias calculadas no contexto dos problemas alvo - Comunicação (67,74%) e Motivação (66,67%) - são ligeiramente inferiores à relevância geral da prática. Este fato pode ser parcialmente explicado pelo comentário do especialista E27, que enfatiza a necessidade de uma comunicação clara e transparente sobre o uso dos dados sensíveis coletados pelos KPIs. A preocupação com a perda de confiança e o risco de microgerenciamento ressalta a importância de uma implementação cuidadosa e responsável dos KPIs.

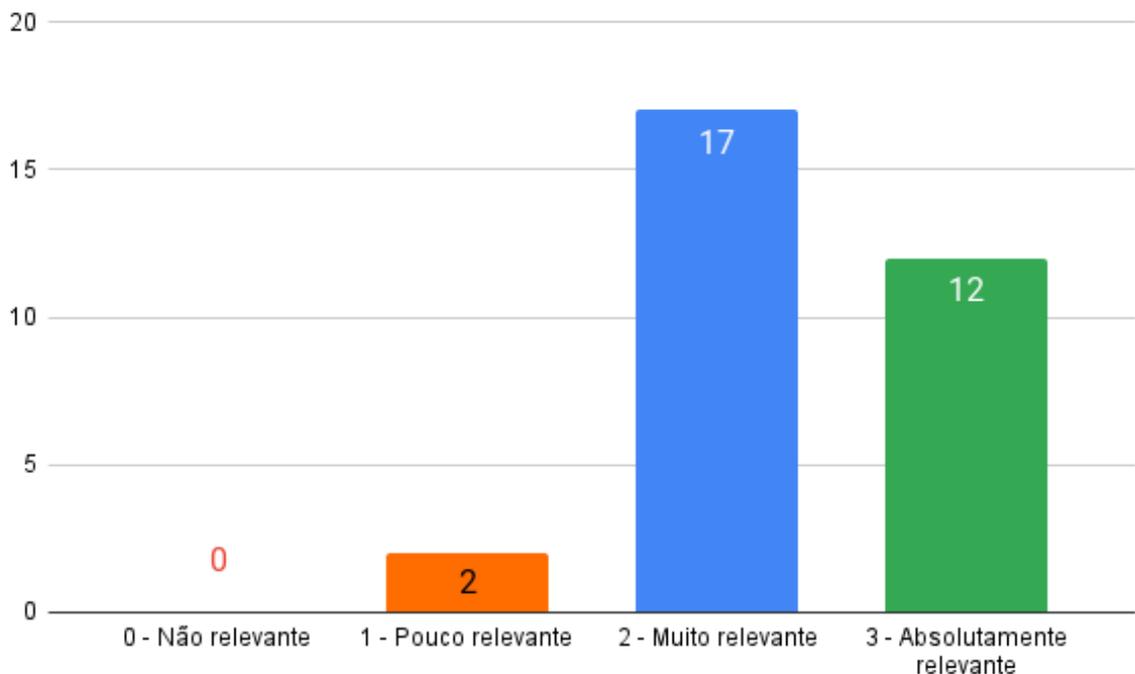


Figura 5.35 – Qual a relevância da prática 016 - Usar indicadores para monitorar a performance do time de desenvolvimento de software para gestão de pessoas em projetos de desenvolvimento de software?

Aqui vejo que é muito importante detalhar para que fim serão usados alguns dos dados mais sensíveis ao time para garantir não perder confiança e não criar um ambiente de microgerenciamento. (Especialista E27)

No contexto da Comunicação, enquanto os KPIs facilitam a comunicação de progresso e metas, eles podem não abordar diretamente os desafios associados ao tamanho do time, escolha das ferramentas de comunicação, ou o overhead de comunicação. Em relação à Motivação, apesar dos KPIs poderem influenciar positivamente a motivação ao estabelecer metas claras e monitorar o progresso, eles não são suficientes por si só para abordar fatores como autonomia, equilíbrio entre vida pessoal e trabalho, ou a qualidade do trabalho. Por outro lado, no que tange ao Foco, a prática tem uma relevância quase tão alta quanto a média geral (75,27%), indicando que o monitoramento estruturado através de KPIs pode ser particularmente eficaz em auxiliar desenvolvedores a manterem-se concentrados em tarefas específicas, minimizando as distrações e otimizando a produtividade.

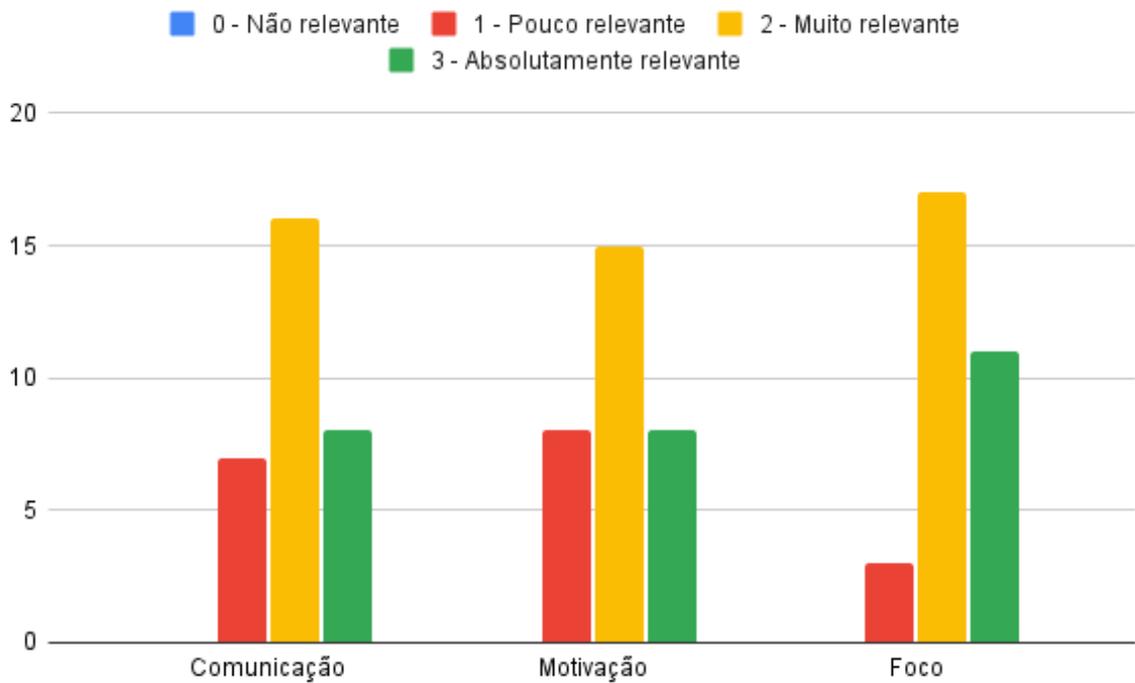


Figura 5.36 – Qual a relevância da prática 016 - Usar indicadores para monitorar a performance do time de desenvolvimento de software para resolver os problemas-alvo identificados?

5.3 Discussão

Após a análise e discussão individual das respostas em cada prática, na presente seção, é discutido os resultados de forma geral, comparando a relevância das práticas entre si. A Tabela 5.3 ordena de acordo com a relevância geral calculada para cada uma das práticas considerando as respostas dos especialistas.

Tabela 5.3 – Ranking de práticas ordenadas por relevância

Ranking	Prática	Relevância
1	P007 - Prover feedback ao time	95.70%
2	P004 - Criar canais e promover a comunicação aberta	90.32%
3	P010 - Praticar gestão à vista	86.02%
4	P002 - Avaliar traços de personalidade e soft skills	84.95%
5	P005 - Criar cronograma mínimo	84.95%
6	P006 - Criar job description para contratação	83.87%
7	P008 - Definir tamanho do time	83.87%
8	P001 - Usar práticas ágeis	82.80%
9	P003 - Executar checklist de onboarding	81.72%
10	P014 - Organizar treinamentos	79.57%
11	P009 - Desafiar time a aprender novas habilidades	78.49%
12	P011 - Identificar e gerenciar competências do time	78.49%
13	P012 - Montar plano de carreira e plano de sucessão	78.49%
14	P016 - Usar indicadores para monitorar a performance do time de desenvolvimento de software	77.42%
15	P013 - Montar time heterogêneo	74.19%
16	P015 - Rotacionar time	68.82%

Analisando a Tabela 5.3 e comparando a relevância por prática, pode-se afirmar, por exemplo, que antes de preparar uma estratégia de rotação de time, o gestor deve assegurar que o time esteja recebendo feedbacks adequados e que possua canais abertos de comunicação.

É importante ressaltar que todas as práticas são relevantes, e possuem embasamento na literatura recente que justificam a sua implementação. A presente análise tem como objetivo indicar um direcionador de prioridades. Conforme já discutido anteriormente, cada projeto tem suas características próprias, cabe ao gestor, tendo conhecimento de quais são as melhores práticas identificar quais devem ser implementadas no seu projeto.

Tabela 5.4 – Relevância geral por prática e relevância para cada problema associado

ID	Prática	Relevância geral	Comunicação	Motivação	Habilidades e conhecimento técnico	Aspectos geográficos	Comprometimento	Satisfação	Estabilidade do time	Foco	Autonomia
P001	Usar práticas ágeis	82.8	79.5	56.5			73.1	64.5		79.5	76.4
P002	Avaliar traços de personalidade e soft skills	84.9						72.0	66.7		
P003	Executar checklist de onboarding	81.7			64.5						70.9
P004	Criar canais e promover a comunicação aberta	90.3	90.3			66.7					
P005	Criar cronograma mínimo	84.9							75.3		
P006	Criar job description para contratação	83.8						73.3	74.2		
P007	Prover feedback ao time	95.7	88.2					90.3			
P008	Definir tamanho do time	83.8	67.8					77.4			
P009	Desafiar time a aprender novas habilidades	78.4		83.9	87.1			78.5			
P010	Praticar gestão à vista	86.0	85.0			57.0	79.5			76.3	72.0
P011	Identificar e gerenciar competências do time	78.5			84.9						
P012	Montar plano de carreira e plano de sucessão	78.5		90.3	75.3						
P013	Montar time heterogêneo	74.2			67.7						54.8
P014	Organizar treinamentos	79.6			84.9						
P015	Rotacionar time	68.8			80.6				51.6		65.6
P016	Usar indicadores para monitorar a performance do time de desenvolvimento de software	77.4	67.7	66.7						75.2	

Considerando as práticas correlacionadas com cada um dos problemas-alvo, a Tabela 5.4 sumariza a relevância de cada prática no contexto de cada um dos problemas. É possível, também, selecionar qual prática tem a maior relevância em cada contexto. A Tabela 5.5 lista problema-alvo, prática mais relevante naquele contexto e relevância calculada.

Com base nos dados apresentados, um gestor que tenha diagnosticado que um de seus projetos de desenvolvimento de software precisa de ações para tratamento de um dos problemas-alvo listados, pode facilmente identificar qual a prática com maior relevância nesse contexto, e priorizar a sua análise e implementação.

Tabela 5.5 – Práticas com maior relevância para cada tipo de problema-alvo

Problema-alvo	Prática (ID)	Relevância
Aspectos geográficos	Criar canais e promover a comunicação aberta (P004A2)	66.67%
Autonomia	Usar práticas ágeis (P001A6)	76.34%
Comprometimento	Praticar gestão à vista (P010A3)	79.57%
Comunicação	Criar canais e promover a comunicação aberta (P004A1)	90.32%
Estabilidade do time	Criar cronograma mínimo (P005A1)	75.27%
Foco	Usar práticas ágeis (P001A5)	79.57%
Habilidades e conhecimentos técnicos	Desafiar time a aprender novas habilidades (P009A3)	87.10%
Motivação	Montar plano de carreira e plano de sucessão (P012A2)	90.32%
Satisfação	Prover feedback ao time (P007A2)	90.32%

A implementação de uma prática deve ser sempre precedida de análise por parte do gestor para garantir que aquela prática realmente é a mais aderente à realidade do seu projeto. A listagem e relevância por prática devem ser utilizadas como direcionador para acelerar o processo, porém não substitui a análise das outras práticas correlacionadas com o problema-alvo.

É notável que algumas práticas tenham maior destaque, analisando a 5.5 identifica-se que algumas práticas aparecem como mais relevantes para mais de um problema-alvo. A Tabela 5.6 descreve a quantidade de problemas-alvo que cada uma das prática foi identificada como a de maior relevância para resolução de problemas.

Tabela 5.6 – Quantidade de problemas alvo

Prática	Quantidade de problemas alvo
P004 - Criar canais e promover a comunicação aberta	2
P001 - Usar práticas ágeis	2
P010 - Praticar gestão à vista	1
P009 - Desafiar time a aprender novas habilidades	1
P005 - Criar cronograma mínimo	1
P012 - Montar plano de carreira e plano de sucessão	1
P007 - Prover feedback ao time	1

Em virtude da ocorrência duplicada para algumas práticas, apenas sete aparecem nessa listagem, deixando nove práticas de fora.

Cruzando as informações presentes na Tabela 5.3 e Tabela 5.6, observa-se que as práticas P004 e P001, estão em primeiro lugar quando analisamos a quantidade de problemas alvo que devem ser priorizadas, enquanto são quarta e oitava, respectivamente, no ranking de relevância geral entre práticas. Da mesma forma, a prática P012, foi identificada como a mais relevante para problemas de motivação, porém no ranking geral de relevância de práticas é listada apenas como a décima terceira. Com base nisso, é possível afirmar que a simples análise de relevância geral de uma prática na opinião dos especialistas, não é suficiente para embasar a escolha da implementação de uma prática por parte do gestor.

Ao selecionar as três práticas (top 3) com maior relevância, superiores a > 85%, e identificar os problemas-alvo correlacionados de cada uma, temos a Tabela 5.7:

Tabela 5.7 – Problemas-alvo correlacionados com as 3 práticas mais relevantes

Prática	Problema-alvo (Correlação)	Relevância
P007 - Prover feedback ao time	Satisfação (P007A2)	90.32%
P007 - Prover feedback ao time	Comunicação (P007A1)	88.17%
P004 - Criar canais e promover a comunicação aberta	Comunicação (P004A1)	90.32%
P004 - Criar canais e promover a comunicação aberta	Aspectos geográficos (P004A2)	66.67%
P010 - Praticar gestão à vista	Comunicação (P010A1)	84.95%
P010 - Praticar gestão à vista	Comprometimento (P010A3)	79.57%
P010 - Praticar gestão à vista	Foco (P010A4)	76.34%
P010 - Praticar gestão à vista	Autonomia (P010A5)	72.04%
P010 - Praticar gestão à vista	Aspectos geográficos (P010A2)	56.99%

Destaca-se que o problema-alvo de comunicação é identificado como correlato a todas as práticas do *top 3* no ranking de relevância geral, em todos os casos com alto grau de relevância. Além disso, uma simples análise das práticas em si, evidenciam a alta relação com comunicação e a significativa ênfase atribuída pelos especialistas a esse tipo de desafio.

Os outros problemas-alvo identificados e correlacionados com as práticas do *top 3* no ranking de relevância geral incluem aspectos geográficos, comprometimento, foco, autonomia e satisfação. Destas, questões ligadas a aspectos geográficos são as únicas mencionadas mais de uma vez, sugerindo também uma alta importância pelos especialistas a este tipo de desafio em projetos de desenvolvimento de software.

Três problemas-alvo não apresentam correlação com as práticas do *top 3* no ranking de relevância geral: Estabilidade do time, habilidades e conhecimentos técnicos e motivação. Nota-se, contudo, que a estabilidade do time está associada à quarta, quinta e sexta práticas mais relevantes. Apesar de não figurar no *top 3*, este problema ainda pode ser apontado como de grande relevância na opinião dos especialistas. Quanto à motivação, ela surge pela primeira vez na prática P001 - Uso de práticas ágeis, classificada como a oitava em relevância geral. Por outro lado, habilidades e conhecimentos técnicos são mencionados pela primeira vez em correlação com a nona prática mais relevante, P003 - Executar checklist de onboarding. Esse último ponto sugere que, na visão dos especialistas, questões relacionadas a habilidades técnicas (*hard skills*) podem ser secundárias, enquanto maior relevância é atribuída aos aspectos ligados a habilidades interpessoais (*soft skills*).

Por fim, ao analisar as práticas consideradas não relevantes nas opiniões dos especialistas, não foi possível identificar um perfil específico em termos de área de atuação, experiência profissional ou formação acadêmica que possa sugerir um contexto onde estas práticas sejam

consideradas menos relevantes.

6 CATÁLOGO DE PRÁTICAS

Nesta seção, apresentamos o catálogo de práticas validadas por especialistas. Na primeira parte, detalhamos o modelo utilizado para apresentar as práticas no catálogo. Este modelo, exibido como um *template* na Tabela 6.1, contém campos específicos: descrição da prática, metodologia utilizada, benefícios, desafios e exemplos de aplicação. O objetivo deste *template* é proporcionar uma estrutura clara para facilitar a compreensão e aplicação das práticas.

ID	Identificador único da prática.
Nome	Nome da prática.
Descrição	Uma breve descrição sobre a prática.
Problema alvo (relevância)	Lista de problemas que podem ser endereçados pelo emprego da prática. Entre parêntesis o grau de relevância da prática no contexto desse problema de acordo com o survey.
Perspectiva da indústria	Discussão de aspectos relevantes da prática a partir da perspectiva dos especialistas.
Relevância geral	Valor percentual que denota o grau de relevância da prática de acordo com o survey.
Referências	Lista de todas as referências que contribuíram para construção da prática.

Tabela 6.1 – Modelo utilizado para descrição das práticas.

Em seguida, introduzimos o catálogo de práticas, validado por especialistas da área. Ele é uma ferramenta útil para pesquisadores e profissionais envolvidos em projetos de desenvolvimento de software, incluindo desenvolvedores, gerentes de projeto e consultores de garantia de qualidade.

ID	P001
Nome	Usar práticas ágeis
Descrição	O emprego de métodos ágeis, como XP e Scrum, promove a eficiência e a adaptabilidade do projeto. A implementação dessas práticas pode ser feita através de treinamentos em métodos ágeis para a adoção de uma mentalidade ágil.
Problema alvo (relevância)	<ul style="list-style-type: none"> • P001A1 - Comunicação (79,57%) • P001A2 - Motivação (56,59%) • P001A3 - Comprometimento (73,12%) • P001A4 - Satisfação (64,52%) • P001A5 - Foco (79,57%) • P001A6 - Autonomia (76,43%)
Perspectiva da indústria	De acordo com os especialistas, essa é a prática mais relevante para problemas relacionados a Foco e Autonomia . Embora a maioria dos participantes tenham indicado algum nível de relevância da prática, houve divergências, com pelo menos uma pessoa, considerando-a como não relevante. Um dos especialistas ressalta que a comunicação é a área que mais se beneficia diretamente do uso de práticas ágeis. Isso enfatiza a ideia de que, no coração das práticas ágeis, está a colaboração e a comunicação eficaz entre os membros da equipe e as partes interessadas.
Relevância geral	82,80%
Referências	Destefanis et al. (2016), Britto, Smite e L. Damm (2016), Iqbal, Omar e Yasin (2019), Ramirez-Mora e Oktaba (2017), Franca, Silva and Sharp (2020), Qahtani (2020), Bass (2016), Shameem et al. (2020), Tam et al. (2020) e Vega-Velazquez, Garcia-Najera e Cervantes (2018)

Tabela 6.2 – Prática 001 - Usar práticas ágeis.

ID	P002
Nome	Avaliar traços de personalidade e soft skills
Descrição	<p>Avaliar traços de personalidade e <i>soft skills</i> garante a correta atribuição de membros a projetos, considerando as especificidades técnicas e a capacidade dos indivíduos. Isso aumenta a produtividade e a satisfação no trabalho e pode ser implementado através de:</p> <ol style="list-style-type: none"> 1. Questionários de personalidade: Questionário que consiste em perguntas para avaliar diferentes traços de personalidade, como extroversão, neuroticismo, abertura a novas experiências, amabilidade e conscienciosidade. Exemplos: Big Five Inventory (BFI) e Myers-Briggs Type Indicator (MBTI). A aplicação de testes psicométricos requer o acompanhamento e interpretação de um profissional treinado como psicólogo ou outros profissionais especialistas; 2. Entrevistas comportamentais estruturadas: Entrevistas projetadas para avaliar como um candidato se comportaria em situações hipotéticas em que o candidato é convidado a responder com base em sua experiência prévia. As respostas ajudam avaliar traços como liderança, trabalho em equipe e resolução de problemas; 3. Testes de desempenho: Testes que consistem na realização de tarefas específicas como escrever código de computador. A forma com que o candidato aborda a tarefa fornece informações sobre a capacidade de trabalhar sob pressão ou atenção aos detalhes; 4. Observação direta: Observar o comportamento do profissional em situações reais.
Problema alvo (relevância)	<ul style="list-style-type: none"> • P002A1 - Estabilidade do time (66,67%) • P002A2 - Satisfação (72,04%)
Perspectiva da indústria	Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Enquanto testes e observações podem fornecer <i>insights</i> valiosos sobre um indivíduo, o comentário de um dos especialistas ressalta que é preciso reconhecer a complexidade das relações humanas e que a dinâmica do grupo pode ser influenciadas por uma combinação de fatores e não apenas pelas características individuais.
Relevância geral	84,95%
Referências	Caulo et al. (2021b), Cardenas-Castro, Julio e Rodriguez (2019), Meyer, Zimmermann e Fritz (2017), Stylianou e Andreou (2016), Yilmaz et al. (2017), Vishnubhotla, Mendes e Lundberg (2020), Anderson et al. (2018), Licorish e MacDonell (2021), Fritzsich et al. (2023) e Cunha et al. (2021)

Tabela 6.3 – Prática 002 - Avaliar traços de personalidade e soft skills.

ID	P003
Nome	Executar checklist de onboarding
Descrição	A criação de <i>checklists</i> para <i>onboarding</i> auxilia na padronização e garantia da completa integração de novos membros ao time, reduzindo o tempo de aclimatação e aumentando a eficiência. Essa prática pode ser implementada com a criação de um <i>checklist</i> que inclua todas as etapas necessárias para a integração de um novo membro.
Problema alvo (relevância)	<ul style="list-style-type: none"> • P003A1 - Habilidades e conhecimentos técnicos (64,52%) • P003A2 - Autonomia (70,97%)
Perspectiva da indústria	Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Nos comentários adicionais feitos pelos especialistas, um deles argumenta que o <i>onboarding</i> estruturado não necessariamente influencia as habilidades ou o conhecimento técnico inerente do profissional, enquanto o segundo ressalta que a habilidade de um indivíduo em encontrar soluções de forma independente é beneficiada pelo <i>onboarding</i> .
Relevância geral	81,72%
Referências	Britto et al. (2019), Britto, Smite e L. Damm (2016) e Britto, Smite, L.-O. Damm et al. (2020)

Tabela 6.4 – Prática 003 - Executar checklist de onboarding.

ID	P004
Nome	Criar canais e promover a comunicação aberta
Descrição	A manutenção de uma comunicação fluida e o estímulo a evitar protocolos fixos contribuem para o fluxo de informação natural e produtivo entre os membros da equipe. Isso reduz mal-entendidos e aumenta a eficiência da equipe. Pode ser implementado através de reuniões regulares, uso de ferramentas de comunicação online e promoção de um ambiente aberto e inclusivo.
Problema alvo (relevância)	<ul style="list-style-type: none"> • P004A1 - Comunicação (90,32%) • P004A2 - Aspectos geográficos (66,67%)
Perspectiva da indústria	De acordo com os especialistas, essa é a prática mais relevante para problemas relacionados à Comunicação e Aspectos geográficos . Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Dois especialistas enfatizaram a relevância da prática para times remotos distribuídos geograficamente.
Relevância geral	90,32%
Referências	Dangmei (2017), Fatema e Kazi Sakib (2017), Bass et al. (2018), X.-N. Shen et al. (2018), Xia, D. Lo et al. (2017), Hidayati, Budiardjo e Purwandari (2020) e Ramirez-Mora e Oktaba (2017)

Tabela 6.5 – Prática 004 - Criar canais e promover a comunicação aberta.

ID	P005
Nome	Criar cronograma mínimo
Descrição	A criação de um cronograma mínimo equilibra tempo e custo, auxiliando na execução eficaz do projeto. Isso ajuda a equipe a se concentrar nas tarefas mais importantes e a evitar atrasos. O planejamento antecipado da alocação dos profissionais auxilia na definição da composição do time, minimizando a necessidade de mudanças no decorrer do projeto. A alocação do recurso correto para a atividade pode considerar fatores individuais (interesse pessoal, atitude, confiança, experiência, conhecimento, papel desempenhado) e critérios do projeto (tipo de projeto, custo, criticidade, risco, complexidade).
Problema alvo (relevância)	<ul style="list-style-type: none"> • P005A1 - Estabilidade do time (75,27%)
Perspectiva da indústria	De acordo com os especialistas, essa é a prática mais relevante para problemas relacionados à Estabilidade do time . Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Dois especialistas enfatizaram, em seus comentários, a importância da prática.
Relevância geral	84,95%
Referências	Maenhout e Vanhoucke (2016), Song et al. (2020), Chiang e Lin (2020), Nigar (2017), X. Shen, Guo e Li (2020), Vega-Velazquez, Garcia-Najera e Cervantes (2018), Paredes-Valverde, Pilar Salas-Zárate et al. (2018), Zapotecas-Martinez, Garcia-Nájera e Cervantes (2020) e Mamatha e Suma (2021)

Tabela 6.6 – Prática 005 - Criar cronograma mínimo.

ID	P006
Nome	Criar job description para contratação
Descrição	Descrições de trabalho claras evitam contratações incorretas, garantindo a seleção de profissionais adequados para cada função. Isso economiza tempo e recursos na contratação e aumenta a satisfação do funcionário. Pode ser implementado com a ajuda de especialistas em RH e <i>feedback</i> dos membros da equipe.
Problema alvo (relevância)	<ul style="list-style-type: none"> • P006A1 - Estabilidade do time (74,19%) • P006A2 - Satisfação (75,27%)
Perspectiva da indústria	Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Um dos especialistas complementa que a contratação de profissionais também se beneficia da aplicação de testes e validações técnicas.
Relevância geral	83,87%
Referências	Nastiti e Setyohadi (2020) e Fritzsich et al. (2023)

Tabela 6.7 – Prática 006 - Criar job description para contratação.

ID	P007
Nome	Prover feedback ao time
Descrição	<p>Prover <i>feedback</i> faz parte do gerenciamento efetivo do time de projeto, auxiliando na melhoria contínua dos membros da equipe, aumenta a moral da equipe e a qualidade do trabalho. É importante a existência de um ambiente de confiança e respeito mútuo, em que o <i>feedback</i> seja visto como oportunidade de crescimento e aprendizado, em vez de crítica pessoal. Pode ser implementado através de reuniões de <i>feedback</i> regulares e sistemas de avaliação entre pares.</p> <p>Algumas técnicas comuns para estruturar um <i>feedback</i> adequado incluem: ser específico e objetivo, fornecer exemplos concretos, enfatizar o comportamento em vez da personalidade, usar uma linguagem clara e direta, e fornecer sugestões construtivas para melhorias futuras.</p> <p>Qualquer membro da equipe pode fornecer <i>feedback</i>, desde que seja feito de forma construtiva e respeitosa, com o objetivo de ajudar a equipe a melhorar.</p>
Problema alvo (relevância)	<ul style="list-style-type: none"> • P007A1 - Comunicação (88,17%) • P007A2 - Satisfação (90,32%)
Perspectiva da indústria	<p>De acordo com os especialistas, essa é a prática mais relevante para problemas relacionados à Satisfação.</p> <p>Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Um dos especialistas sugere que além dos problemas-alvo inicialmente identificados, a motivação também pode ser relacionada a essa prática.</p>
Relevância geral	95,70%
Referências	Britto, Smite e L. Damm (2016), Zaouga, Rabai e Alalyani (2019), Cunha et al. (2016) e Dzvoniyar e Bruegge (2018)

Tabela 6.8 – Prática 007 - Prover feedback ao time.

ID	P008
Nome	Definir tamanho do time
Descrição	<p>Definir o tamanho ideal do time é uma tarefa do gestor, que deve equilibrar a quantidade de membros e a eficiência na entrega do software. O tamanho ideal de uma equipe pode depender de vários fatores, incluindo a complexidade do projeto, a experiência dos membros da equipe e a natureza das tarefas a serem realizadas.</p> <p>Em geral, o aumento do tamanho da equipe pode aumentar a eficiência até certa medida, mas, depois disso, o custo de coordenação aumenta e a eficiência começa a diminuir. O custo de coordenação cresce exponencialmente na proporção que os caminhos de comunicação entre os membros do time aumenta.</p>
Problema alvo (relevância)	<ul style="list-style-type: none"> • P008A1 - Comunicação (67,74%) • P008A2 - Satisfação (77,42%)
Perspectiva da indústria	Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Nenhum comentário adicional foi feito.
Relevância geral	83,87%
Referências	Wang, Huang e Liu (2018) e Scott, Charkie e Pfahl (2020)

Tabela 6.9 – Prática 008 - Definir tamanho do time.

ID	P009
Nome	Desafiar time a aprender novas habilidades
Descrição	O gestor deve incentivar o time a adquirir novas habilidades, promovendo o crescimento profissional e a adaptabilidade da equipe. Isso aumenta a competitividade e a flexibilidade da equipe. Pode ser implementado através de treinamentos, <i>workshops</i> e incentivos para a educação contínua.
Problema alvo (relevância)	<ul style="list-style-type: none"> • P009A1 - Motivação (83,87%) • P009A2 - Satisfação (78,49%) • P009A3 - Habilidades e conhecimentos técnicos (87,10%)
Perspectiva da indústria	De acordo com os especialistas, essa é a prática mais relevante para problemas relacionados à Habilidades e conhecimentos técnicos . Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Nenhum comentário adicional foi feito.
Relevância geral	78,49%
Referências	R. E. S. Santos, F. Q. B. d. Silva et al. (2016), Cardenas-Castro, Julio e Rodriguez (2019) e Dzvonyar e Bruegge (2018)

Tabela 6.10 – Prática 009 - Desafiar time a aprender novas habilidades.

ID	P010
Nome	Praticar gestão à vista
Descrição	<p>A disponibilização de indicadores de maneira visível para todo o time, seja em quadro físico ou ferramenta online, promove a transparência e o alinhamento das metas. Isso aumenta a consciência da equipe sobre o status do projeto e permite que o próprio time identifique rapidamente os problemas e as áreas que precisam de mais atenção, permitindo que eles sejam abordados de forma proativa antes que se tornem um obstáculo para o progresso do projeto.</p> <p>Pode ser implementado com o uso de quadros de gestão à vista, dashboards online e reuniões de status regulares.</p> <p>Recomenda-se que as medidas sejam feitas como equipe única, para que "não possa haver vencedores ou perdedores individuais" e todos trabalhem em direção ao mesmo objetivo.</p>
Problema alvo (relevância)	<ul style="list-style-type: none"> • P010A1 - Comunicação (84,95%) • P010A2 - Aspectos geográficos (56,99%) • P010A3 - Comprometimento (79,57%) • P010A4 - Foco (76,32%) • P010A5 - Autonomia (72,04%)
Perspectiva da indústria	<p>De acordo com os especialistas, essa é a prática mais relevante para problemas relacionados a Comprometimento.</p> <p>Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Um dos especialistas ressalta a necessidade de planejamento para o emprego da prática, alertando sobre o risco de focar em métricas que podem desviar a atenção dos desafios estratégicos. Embora a prática seja de grande importância, com impactos em diversos tipos de problema, sua eficácia depende da qualidade e relevância das informações expostas.</p>
Relevância geral	86,02%
Referências	Destefanis et al. (2016), Cunha et al. (2016) e Shameem et al. (2020)

Tabela 6.11 – Prática 010 - Praticar gestão à vista.

ID	P011
Nome	Identificar e gerenciar competências do time
Descrição	Mapear os conhecimentos necessários e os existentes no time permite um gerenciamento efetivo das competências, contribuindo para a eficiência e o desenvolvimento do projeto. Essa prática pode ser implementada por meio de avaliações de competência de forma periódica e planos de treinamento personalizados.
Problema alvo (relevância)	<ul style="list-style-type: none"> • P011A1 - Habilidades e conhecimentos técnicos (84,95%)
Perspectiva da indústria	Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Nenhum comentário adicional foi feito.
Relevância geral	78,49%
Referências	Dangmei (2017), Hidayati, Budiardjo e Purwandari (2020), Paredes-Valverde, Salas-Zárate et al. (2018), Angelis (2019), Meyer, Zimmermann e Fritz (2017), Zaouga, Rabai e Alalyani (2019), Paredes-Valverde, Pilar Salas-Zárate et al. (2018) e Bakanova e Shikov (2020)

Tabela 6.12 – Prática 011 - Identificar e gerenciar competências do time.

ID	P012
Nome	Montar plano de carreira e plano de sucessão
Descrição	<p>A implementação de planos de carreira e sucessão, aliados a treinamentos, ajuda a motivar e reter talentos, além de garantir a continuidade do projeto quando um profissional deixa a empresa ou muda de posição.</p> <p>Essa prática pode ser implementada por meio de reuniões de carreira individuais e planos de desenvolvimento personalizados. A prática inclui: identificar potenciais sucessores, fornecer treinamento, oferecer oportunidades de mentoria, criação de plano de transição e realização de testes de sucessão.</p> <p>É recomendado que a empresa identifique mais de um potencial sucessor para cada posição crítica, para garantir que haja uma opção de backup caso o primeiro candidato não esteja disponível ou não seja adequado para a posição. Além disso, é importante que a empresa avalie regularmente seus planos de sucessão e faça ajustes conforme necessário, para garantir que estejam atualizados e alinhados com as necessidades da empresa e dos funcionários.</p>
Problema alvo (relevância)	<ul style="list-style-type: none"> • P012A1 - Habilidades e conhecimentos técnicos (75,27%) • P012A2 - Motivação (90,32%)
Perspectiva da indústria	<p>De acordo com os especialistas, essa é a prática mais relevante para problemas relacionados à Motivação.</p> <p>Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Nenhum comentário adicional foi feito.</p>
Relevância geral	78,49%
Referências	Nicolaescu et al. (2020) e Trinkenreich et al. (2023)

Tabela 6.13 – Prática 012 - Montar plano de carreira e plano de sucessão.

ID	P013
Nome	Montar time heterogêneo
Descrição	<p>Formar um time heterogêneo permite que a abordagem de problemas de diferentes maneiras e, para os quais, encontrem soluções criativas e eficazes. Isso pode ser implementado através de uma política de recrutamento e seleção que valoriza a diversidade, o que pode incluir diversidade de gênero, raça, etnia, orientação sexual, idade, entre outros.</p> <p>Cabe ao gestor criar um ambiente que incentiva a colaboração entre membros da equipe e que promova a troca de ideias e conhecimentos, permitindo que os membros da equipe aprendam uns com os outros e trabalhem juntos para alcançar objetivos comuns.</p>
Problema alvo (relevância)	<ul style="list-style-type: none"> • P013A1 - Habilidades e conhecimentos técnicos (67,74%) • P013A2 - Autonomia (54,84%)
Perspectiva da indústria	Embora a maioria dos participantes tenham indicado algum nível de relevância da prática, houve divergências, com três participantes considerando-a como não relevante. Nenhum comentário adicional foi feito.
Relevância geral	74,19%
Referências	Xia, D. Lo et al. (2017), Canedo e G. Santos (2019), Nastiti e Setyohadi (2020) e Cunha et al. (2021)

Tabela 6.14 – Prática 013 - Montar time heterogêneo.

ID	P014
Nome	Organizar treinamentos
Descrição	<p>Fornecer treinamento constante aos profissionais melhora suas habilidades e aumenta a eficiência do projeto. Isso pode ser implementado por meio de programas de treinamento internos ou externos e aprendizado online.</p> <p>Recomendam-se treinamentos compostos por uma combinação de treinamento formal, aprendizado baseado em trabalho real e mentoria de desenvolvedores mais experientes.</p> <p>Profissionais sênior podem ser uma boa opção para ministrar treinamentos, mas não necessariamente a única. A empresa pode incentivar todos seus profissionais a se tornarem mentores, oferecendo oportunidades de desenvolvimento de carreira e reconhecimento por seu trabalho. A empresa também pode fornecer treinamento e suporte para os mentores, para que eles possam desempenhar suas funções com eficácia.</p> <p>A criação de um ambiente de trabalho que valorize a aprendizagem e o desenvolvimento contínuo incentivam as pessoas a participarem dos treinamentos, e é comum que as empresas emitam certificados para os funcionários que concluem os programas de treinamento. Isso pode ajudar a reconhecer o esforço e o desempenho dos funcionários, além de fornecer uma prova tangível de suas habilidades e conhecimentos adquiridos. No entanto, a decisão de emitir certificados após a conclusão do treinamento depende das políticas e práticas da empresa.</p>
Problema alvo (relevância)	<ul style="list-style-type: none"> • P014A1 - Habilidades e conhecimentos técnicos (84,95%)
Perspectiva da indústria	Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Nenhum comentário adicional foi feito.
Relevância geral	79,57%
Referências	Fatema e K. Sakib (2017), Shahzad, Xiu e Shahbaz (2017) e Britto, Smite, L.-O. Damm et al. (2020)

Tabela 6.15 – Prática 014 - Organizar treinamentos.

ID	P015
Nome	Rotacionar time
Descrição	<p>A rotação de time pode ser realizada de duas maneiras:</p> <ol style="list-style-type: none"> 1. A rotação de tarefas (<i>job-to-job</i>) dentro do mesmo time ou projeto permite aos membros adquirirem uma variedade de habilidades e conhecimentos através da mudança de papéis entre membros do mesmo time. Pode ser implementada através de um programa de rotação de tarefas bem estruturado. 2. A rotação entre projetos (<i>project-to-project</i>) permite aos membros trazerem novas perspectivas e aprender com diferentes contextos. Pode ser implementado ao designar membros da equipe para novos projetos em um ciclo regular, o que também ajuda a evitar a estagnação e promover a inovação. <p>A frequência ideal da rotação de trabalho pode variar dependendo do contexto específico e das necessidades da organização, ela deve ser frequente o suficiente para fornecer aos funcionários novos desafios e oportunidades de aprendizado, mas não tão frequente que interrompa o trabalho ou cause instabilidade.</p>
Problema alvo (relevância)	<ul style="list-style-type: none"> • P015A1 - Estabilidade do time (51,61%) • P015A2 - Habilidades e conhecimentos técnicos (80,65%) • P015A3 - Autonomia (65,59%)
Perspectiva da indústria	Embora a maioria dos participantes tenham indicado algum nível relevância da prática, houve divergências, com pelo menos uma pessoa, considerando-a como não relevante. Um dos especialistas enfatiza o cuidado na implantação da prática, podendo evidenciar problemas de relacionamento ou de natureza técnica.
Relevância geral	68,82%
Referências	R. E. S. Santos, F. Q. B. d. Silva et al. (2016), R. E. S. Santos, F. Silva e C. V. C. Magalhães (2016), Magalhães (2017), Santos (2017), Govindaras et al. (2023), R. E. S. Santos (2017) e Dzvonyar e Bruegge (2018)

Tabela 6.16 – Prática 015 - Rotacionar time.

ID	P016
Nome	Usar indicadores para monitorar a performance do time de desenvolvimento de software
Descrição	O uso de indicadores-chave de desempenho ou <i>key performance indicators</i> (KPIs) permite medir e monitorar o desempenho do time, promovendo a melhoria contínua. Essa prática pode ser implementada através do uso de software de rastreamento de projetos e ferramentas de análise de desempenho.
Problema alvo (relevância)	<ul style="list-style-type: none"> • P016A1 - Comunicação (67,74%) • P016A2 - Motivação (66,67%) • P016A3 - Foco (75,27%)
Perspectiva da indústria	Todos os participantes indicaram - em diferentes níveis - que a prática é relevante. Um dos especialistas enfatiza a necessidade de uma comunicação clara e transparente sobre o uso dos dados sensíveis coletados pelos KPIs. A sua implementação deve ser feita de forma cuidadosa e responsável para evitar perda de confiança e o risco de microgerenciamento.
Relevância geral	77,42%
Referências	Oliveira et al. (2016), Cunha et al. (2021) e Nicolaescu et al. (2020)

Tabela 6.17 – Prática 016 - Usar indicadores para monitorar a performance do time de desenvolvimento de software.

7 CONCLUSÃO

Neste capítulo final, resumimos as principais descobertas, abordamos as limitações metodológicas e sugerimos direções para pesquisas futuras, visando contribuir para o avanço do campo de estudo e inspirar trabalhos subsequentes.

7.1 Considerações finais

Este estudo investigou práticas de gestão de pessoas em projetos de desenvolvimento de software, visando elaborar um catálogo de melhores práticas para ser usado como recurso de referência por pesquisadores e profissionais atuantes em projetos de desenvolvimento de software. Esse público inclui, mas não se limita, a desenvolvedores, gerentes de projeto e consultores de garantia de qualidade.

A metodologia do trabalho incluiu a realização de revisão sistemática literária para identificar problemas oriundos da gestão de pessoas em projetos de desenvolvimento de software e quais práticas aplicáveis para solucioná-los. Um catálogo de práticas foi então desenvolvido e submetido à validação de especialistas no campo, por meio de questionários.

Utilizando quatro engenhos de busca, foram inicialmente identificados 2495 estudos primários não duplicados, os quais, após aplicação de critérios de seleção descritos no protocolo de revisão, resultaram em 63 estudos aceitos para extração de dados e sumarização de resultados.

Foram identificados *nove* agrupamentos de problemas: Aspectos geográficos, autonomia, comprometimento, comunicação, estabilidade do time, foco, habilidades e conhecimentos técnicos, motivação e satisfação. Além disso, *oito* grupos de práticas foram estruturadas em *dezesseis* práticas distintas.

As práticas foram validadas por 31 especialistas de diferentes áreas de atuação. Utilizando as respostas dos especialistas foi possível atribuir uma relevância numérica e, a partir disso, classificar as práticas em mais ou menos relevantes.

A análise das respostas dos especialistas revelou *insights* valiosos sobre a relevância da implementação e impacto dessas práticas. Foi possível observar que mesmo as práticas de maior relevância, quando implementadas de maneira inadequada, podem ser ineficazes. Isso reforça a ideia de Kerzner (2006) de que não há consenso entre as empresas quanto a quais práticas são as melhores.

As principais práticas identificadas foram: (1) prover *feedback* ao time, (2) criar canais e

promover a comunicação aberta e (3) praticar gestão à vista, dentre as quais, destaca-se o fato de que todas correlacionam-se de alguma forma com o problema-alvo de comunicação, um tema frequentemente citado na literatura como desafio na gestão de pessoas em projetos de software.

Ao analisar a relevância das práticas, é possível observar que, segundo os especialistas, questões relacionadas a habilidades técnicas (*hard skills*) podem ser secundárias, enquanto maior relevância é atribuída aos aspectos ligados a habilidades interpessoais (*soft skills*).

Os resultados oferecem diretrizes para a implementação de práticas de gestão em projetos de desenvolvimento de software. Cada prática, com suas particularidades, contribui de maneira única para o sucesso do projeto e satisfação da equipe. Contudo, as limitações do estudo, incluindo a variabilidade nas respostas dos especialistas e a especificidade do contexto de software, devem ser consideradas ao generalizar os resultados.

Conclui-se que a gestão eficaz de pessoas em projetos de desenvolvimento de software é composto pela combinação estratégica de várias práticas. O sucesso desses projetos depende não apenas do conhecimento técnico, mas também de uma gestão de pessoas eficaz, que considera comunicação, motivação, satisfação e desenvolvimento pessoal e profissional dos membros da equipe. Assim, o catálogo proposto é um recurso valioso para gestores avaliarem as práticas disponíveis, selecionarem e adaptarem aquelas mais adequadas à sua realidade específica.

7.2 Limitações do estudo

Este estudo apresenta limitações que merecem ser consideradas. A utilização de quatro engines de busca específicos (ACM, IEEE, SciELO, e ScienceDirect) para identificar estudos primários restringiu o escopo das pesquisas incluídas, potencialmente omitindo estudos relevantes publicados fora dessas bases ou em idiomas não abrangidos. Embora essa escolha tenha sido estratégica para focar em fontes reconhecidas por sua qualidade acadêmica e relevância para o campo do desenvolvimento de software, reconhece-se que tal delimitação pode limitar a abrangência das práticas identificadas.

Ademais, a classificação e seleção das práticas de gestão de pessoas, embora embasadas na literatura e validadas por especialistas, foram influenciadas pelas interpretações e critérios dos pesquisadores, podendo não captar todas as nuances em diferentes contextos de desenvolvimento de software.

A visão da indústria, especificamente, é limitada às opiniões dos 31 especialistas que responderam ao questionário de validação, representando uma amostra restrita que pode não re-

fletir a diversidade de experiências e percepções existentes no setor. Além disso, a variabilidade nas respostas dos especialistas sugere uma subjetividade na avaliação da relevância das práticas, e a especificidade do contexto de software pode restringir a generalização dos resultados para outros domínios ou tipos de projetos.

A eficácia das práticas identificadas depende significativamente de sua correta implementação, um aspecto que não foi profundamente explorado neste estudo. Portanto, a aplicação dos resultados requer uma cuidadosa consideração do contexto específico de cada projeto e a adaptabilidade das práticas recomendadas.

7.3 Trabalhos futuros

Como trabalhos futuros, podemos buscar quantificar os benefícios das práticas de gestão de pessoas em projetos de desenvolvimento de software, criando um modelo para medir o impacto dessas práticas em aspectos como a produtividade da equipe, qualidade do software, e satisfação no trabalho. Poderíamos usar indicadores chave de desempenho (KPIs) e realizar estudos de caso em organizações para avaliar o efeito dessas práticas a curto e longo prazo, proporcionando uma base mais sólida para decisões baseadas em dados no campo da gestão de pessoas em projetos de desenvolvimento de software.

REFERÊNCIAS

- Alcântara, Pedro, Edna Canedo e Ruyther Costa (2018). “People Management in Software Agile Development”. Em: *Proceedings of the XIV Brazilian Symposium on Information Systems*. SBSI’18. Caxias do Sul, Brazil: Association for Computing Machinery. ISBN: 9781450365598. DOI: [10.1145/3229345.3229396](https://doi.org/10.1145/3229345.3229396). URL: <https://doi.org/10.1145/3229345.3229396>.
- Anderson, Greg et al. (2018). *The Effect of Software Team Personality Composition on Learning and Performance: Making the "Dream" Team*. DOI: [10.24251/HICSS.2018.059](https://doi.org/10.24251/HICSS.2018.059).
- Angelis, Lefteris (2019). “Statistical Analysis in The Research of Human Factor in Software Engineering: Invited Talk Paper”. Em: *Proceedings of the 9th Balkan Conference on Informatics*. BCI’19. Sofia, Bulgaria: Association for Computing Machinery. ISBN: 9781450371933. DOI: [10.1145/3351556.3351593](https://doi.org/10.1145/3351556.3351593). URL: <https://doi.org/10.1145/3351556.3351593>.
- Bakanova, Anna e Aleksey Nikolaevich Shikov (2020). “The method of the best performer selection based on a competency-based approach”. Em: *SHS Web of Conferences*. Vol. 89. EDP Sciences, p. 03004.
- Bass, Julian M. (2016). “Artefacts and agile method tailoring in large-scale offshore software development programmes”. Em: *Information and Software Technology* 75, pp. 1–16. ISSN: 0950-5849. DOI: <https://doi.org/10.1016/j.infsof.2016.03.001>. URL: <https://www.sciencedirect.com/science/article/pii/S0950584916300350>.
- Bass, Julian M. et al. (2018). “Employee Retention and Turnover in Global Software Development: Comparing In-House Offshoring and Offshore Outsourcing”. Em: *2018 IEEE/ACM 13th International Conference on Global Software Engineering (ICGSE)*, pp. 77–86.
- Beck, Kent (1995). “Design patterns: Elements of reusable object-oriented software”. Em: *IBM Systems Journal* 34.3, p. 544.
- Beck, Kent et al. (2001). *Manifesto for Agile Software Development*. URL: <http://www.agilemanifesto.org/>.
- Britto, Ricardo, Darja Smite e L. Damm (2016). *Experiences from Measuring Learning and Performance in Large-Scale Distributed Software Development*. DOI: [10.1145/2961111.2962636](https://doi.org/10.1145/2961111.2962636).

- Britto, Ricardo, Darja Smite, Lars-Ola Damm et al. (2020). “Evaluating and strategizing the onboarding of software developers in large-scale globally distributed projects”. Em: *Journal of Systems and Software* 169, p. 110699.
- (2019). “Performance Evolution of Newcomers in Large-Scale Distributed Software Projects: An Industrial Case Study”. Em: *2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE)*, pp. 1–11. DOI: [10.1109/ICGSE.2019.00000](https://doi.org/10.1109/ICGSE.2019.00000).
- Burkard, Marcelo Falkowski e Lisandra Manzoni Fontoura (2023). “People Management Problems and Practices in Software Development Projects: A Systematic Literature Review.” Em: *ICEIS (2)*, pp. 179–186.
- Canedo, E. e G. Santos (2019). *Factors Affecting Software Development Productivity: An empirical study*. DOI: [10.1145/3350768.3352491](https://doi.org/10.1145/3350768.3352491).
- Cardenas-Castro, Carol, Jenny Carolina Gil Julio e Paula Rodriguez (2019). “Soft Skills Training: Performance Psychology Applied to Software Development”. Em: *2019 IEEE/ACM 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*, pp. 115–116. DOI: [10.1109/CHASE.2019.00034](https://doi.org/10.1109/CHASE.2019.00034).
- Caulo, Maria et al. (2021a). “Relationships between Personality Traits and Productivity in a Multi-Platform Development Context”. Em: *Evaluation and Assessment in Software Engineering*. EASE 2021. Trondheim, Norway: Association for Computing Machinery, pp. 70–79. ISBN: 9781450390538. DOI: [10.1145/3463274.3463327](https://doi.org/10.1145/3463274.3463327). URL: <https://doi.org/10.1145/3463274.3463327>.
- (2021b). “Relationships between Personality Traits and Productivity in a Multi-Platform Development Context”. Em: *Evaluation and Assessment in Software Engineering*. EASE 2021. Trondheim, Norway: Association for Computing Machinery, pp. 70–79. ISBN: 9781450390538. DOI: [10.1145/3463274.3463327](https://doi.org/10.1145/3463274.3463327). URL: <https://doi.org/10.1145/3463274.3463327>.
- Chiang, Hui Yi e Bertrand M. T. Lin (2020). “A Decision Model for Human Resource Allocation in Project Management of Software Development”. Em: *IEEE Access* 8, pp. 38073–38081. DOI: [10.1109/ACCESS.2020.2975829](https://doi.org/10.1109/ACCESS.2020.2975829).
- Cunha, Felipe et al. (2021). “A decision support system for multiple team formation”. Em: *Anais do I Workshop Brasileiro de Engenharia de Software Inteligente*. SBC, pp. 13–28.

- Cunha, José Adson da et al. (2016). “Towards a substantive theory of decision-making in software project management: Preliminary findings from a qualitative study”. Em: *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, pp. 1–10.
- Curtis, Bill, William E Hefley e Sally A Miller (2009). *People CMM: A framework for human capital management*. Pearson Education.
- Dangmei, Jianguanglung (2017). “People Capability Maturity Model (P-CMM) Facing the Key HR Challenges of MSME in India: A Theoretical Approach”. Em: *Asian Journal of Management*. DOI: [10.5958/2321-5763.2017.00022.1](https://doi.org/10.5958/2321-5763.2017.00022.1).
- Demir, Kadir Alpaslan (2017). “3PR Framework for Software Project Management: People, Process, Product, and Risk”. Em: *Software Project Management for Distributed Computing: Life-Cycle Methods for Developing Scalable and Reliable Tools*. Ed. por Zaigham Mahmood. Cham: Springer International Publishing, pp. 143–170. ISBN: 978-3-319-54325-3. DOI: [10.1007/978-3-319-54325-3_7](https://doi.org/10.1007/978-3-319-54325-3_7). URL: https://doi.org/10.1007/978-3-319-54325-3_7.
- Destefanis, G. et al. (2016). *Software development: do good manners matter?* DOI: [10.7287/PEERJ.PREPRINTS.1515V1](https://doi.org/10.7287/PEERJ.PREPRINTS.1515V1).
- Dzvonyar, Dora e Bernd Bruegge (2018). “Team Composition and Team Factors in Software Engineering: An Interview Study of Project-Based Organizations”. Em: *2018 25th Asia-Pacific Software Engineering Conference (APSEC)*. IEEE, pp. 561–570.
- Fatema, Israt e K. Sakib (2017). *Factors Influencing Productivity of Agile Software Development Teamwork: A Qualitative System Dynamics Approach*. DOI: [10.1109/APSEC.2017.95](https://doi.org/10.1109/APSEC.2017.95).
- Fatema, Israt e Kazi Sakib (2017). “Factors Influencing Productivity of Agile Software Development Teamwork: A Qualitative System Dynamics Approach”. Em: *2017 24th Asia-Pacific Software Engineering Conference (APSEC)*, pp. 737–742. DOI: [10.1109/APSEC.2017.95](https://doi.org/10.1109/APSEC.2017.95).
- Franca, César, Fabio Q. B. da Silva e Helen Sharp (2020). “Motivation and Satisfaction of Software Engineers”. Em: *IEEE Transactions on Software Engineering* 46.2, pp. 118–140. DOI: [10.1109/TSE.2018.2842201](https://doi.org/10.1109/TSE.2018.2842201).
- Fritzsich, Jonas et al. (2023). “Resist the Hype! Practical Recommendations to Cope With Résumé-Driven Development”. Em: *IEEE Software*.

- Fuentes, S.L., A. Frankenberg-Garcia e H. Newstead (2015). *Dicionário Oxford de Português*. Oxford University Press. ISBN: 9780199678129. URL: <https://books.google.com.br/books?id=nG7EoQEACAAJ>.
- García, Félix et al. (2017). “A framework for gamification in software engineering”. Em: *Journal of Systems and Software* 132, pp. 21–40. ISSN: 0164-1212. DOI: <https://doi.org/10.1016/j.jss.2017.06.021>. URL: <https://www.sciencedirect.com/science/article/pii/S0164121217301218>.
- Govindaras, Baskaran et al. (2023). “Sustainable Environment to Prevent Burnout and Attrition in Project Management”. Em: *Sustainability* 15.3, p. 2364.
- Hattie, John e Helen Timperley (2007). “The power of feedback”. Em: *Review of educational research* 77.1, pp. 81–112.
- Hidayati, Anita, E. K. Budiardjo e B. Purwandari (2020). *Hard and Soft Skills for Scrum Global Software Development Teams*. DOI: [10.1145/3378936.3378966](https://doi.org/10.1145/3378936.3378966).
- Iqbal, J., Mazni Omar e A. Yasin (2019). “An Empirical Analysis of the Effect of Agile Teams on Software Productivity”. Em: DOI: [10.1109/ICOMET.2019.8673413](https://doi.org/10.1109/ICOMET.2019.8673413).
- Johnson, Rick (2008). *Six Principles of Effective Team Management*. Acessado: 02/06/2022. URL: <https://ezinearticles.com/?Six-Principles-of-Effective-Team-Management&id=1803062>.
- Kajornboon, Annabel Bhamani (2005). “Using interviews as research instruments”. Em: *E-journal for Research Teachers* 2.1, pp. 1–9.
- Kerzner, Harold (2006). *Gestão de Projetos-: As Melhores Práticas*. Bookman editora.
- Kitchenham, Barbara e Stuart Charters (2007). *Guidelines for performing Systematic Literature Reviews in Software Engineering*. Rel. técn. EBSE 2007-001. <http://www.dur.ac.uk/ebse/resources/reviews-5-8.pdf>. Keele University e Durham University Joint Report.
- Kohl, Karina, Bogdan Vasilescu e Rafael Prikladnicki (2020). “Multitasking Across Industry Projects: A Replication Study”. Em: *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*. New York, NY, USA: Association for Computing Machinery, pp. 93–100. ISBN: 9781450379632. URL: <https://doi.org/10.1145/3387940.3391495>.
- Kula, Elvan et al. (2021). “Factors Affecting On-Time Delivery in Large-Scale Agile Software Development”. Em: *IEEE Transactions on Software Engineering*, pp. 1–1. DOI: [10.1109/TSE.2021.3101192](https://doi.org/10.1109/TSE.2021.3101192).

- Lalsing, Vikash, Somveer Kishnah e Sameerchand Pudaruth (2012). “People factors in agile software development and project management”. Em: *International Journal of Software Engineering & Applications* 3.1, p. 117.
- Licorish, Sherlock A. e Stephen G. MacDonell (2021). *Personality Profiles of Global Software Developers*. DOI: [10.1145/2601248.2601265](https://doi.org/10.1145/2601248.2601265).
- Machuca-Villegas, Liliana et al. (2022). “Perceptions of the human and social factors that influence the productivity of software development teams in Colombia: A statistical analysis”. Em: *Journal of Systems and Software* 192, p. 111408. ISSN: 0164-1212. DOI: <https://doi.org/10.1016/j.jss.2022.111408>. URL: <https://www.sciencedirect.com/science/article/pii/S0164121222001224>.
- Maenhout, Broos e Mario Vanhoucke (2016). “An exact algorithm for an integrated project staffing problem with a homogeneous workforce”. Em: *Journal of Scheduling*. DOI: [10.1007/S10951-015-0443-Z](https://doi.org/10.1007/S10951-015-0443-Z).
- Magalhães, Cleyton de (2017). “Toward understanding work characteristics in software engineering”. Em: *ACM SIGSOFT Software Engineering Notes* 41.6, pp. 1–6.
- Mamatha, R e KG Suma (2021). “A Preliminary Review on Resource Allocation Within a Software Team.” Em: *Turkish Online Journal of Qualitative Inquiry* 12.5.
- Margareth, Ayu Rifanny e Adi Mulyanto (2021). “Designing Project Management Application to Support Developer Productivity using Goal Directed Design Approach”. Em: *2021 International Conference on Data and Software Engineering (ICoDSE)*, pp. 1–6. DOI: [10.1109/ICoDSE53690.2021.9648463](https://doi.org/10.1109/ICoDSE53690.2021.9648463).
- Meyer, André N., Thomas Zimmermann e Thomas Fritz (2017). “Characterizing Software Developers by Perceptions of Productivity”. Em: *2017 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, pp. 105–110. DOI: [10.1109/ESEM.2017.17](https://doi.org/10.1109/ESEM.2017.17).
- Mishra, Amit e Sanjay Misra (nov. de 2010). “People Management in Software Industry: The Key to Success”. Em: *SIGSOFT Softw. Eng. Notes* 35.6, pp. 1–4. ISSN: 0163-5948. DOI: [10.1145/1874391.1874402](https://doi-org.ez47.periodicos.capes.gov.br/10.1145/1874391.1874402). URL: <https://doi-org.ez47.periodicos.capes.gov.br/10.1145/1874391.1874402>.
- Nastiti, Putri e Djoko Budiyanto Setyohadi (2020). “Identification of Critical Factors on Structure and Conduct using SCP Framework: Evidence from Indonesian ICT Industry”. Em:

- 2020 17th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 24–29. DOI: [10.1109/JCSSE49651.2020.9268252](https://doi.org/10.1109/JCSSE49651.2020.9268252).
- Ng, Jun Jie (2018). “Tailoring a Project Management Methodology That Suits One’s Needs”. Em: *IEEE Engineering Management Review* 46.2, pp. 49–54. DOI: [10.1109/EMR.2018.2839670](https://doi.org/10.1109/EMR.2018.2839670).
- Nicolaescu, Sergiu Stefan et al. (2020). “Human capital evaluation in knowledge-based organizations based on big data analytics”. Em: *Future Generation Computer Systems* 111, pp. 654–667. ISSN: 0167-739X. DOI: <https://doi.org/10.1016/j.future.2019.09.048>. URL: <https://www.sciencedirect.com/science/article/pii/S0167739X19306351>.
- Nigar, Natasha (2017). “Model-Based Dynamic Software Project Scheduling”. Em: *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*. ESEC/FSE 2017. Paderborn, Germany: Association for Computing Machinery, pp. 1042–1045. ISBN: 9781450351058. DOI: [10.1145/3106237.3119879](https://doi.org/10.1145/3106237.3119879). URL: <https://doi.org/10.1145/3106237.3119879>.
- Oishi, Sabine Mertens (2003). *How to conduct in-person interviews for surveys*. Sage. DOI: <https://doi.org/10.4135/9781412984416>.
- Oliveira, Edson et al. (2016). “Software Project Managers’ Perceptions of Productivity Factors: Findings from a Qualitative Study”. Em: *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*. ESEM ’16. Ciudad Real, Spain: Association for Computing Machinery. ISBN: 9781450344272. DOI: [10.1145/2961111.2962626](https://doi.org/10.1145/2961111.2962626). URL: <https://doi.org/10.1145/2961111.2962626>.
- Paredes-Valverde, Mario Andrés, María del Pilar Salas-Zárate et al. (2018). “An ontology-based approach with which to assign human resources to software projects”. Em: *Science of Computer Programming* 156, pp. 90–103. ISSN: 0167-6423. DOI: <https://doi.org/10.1016/j.scico.2018.01.003>. URL: <https://www.sciencedirect.com/science/article/pii/S0167642318300042>.
- Paredes-Valverde, Mario Andrés, María del Pilar Salas-Zárate et al. (2018). “An ontology-based approach with which to assign human resources to software projects”. Em: *Science of Computer Programming*. DOI: [10.1016/J.SCICO.2018.01.003](https://doi.org/10.1016/J.SCICO.2018.01.003).

- PMI, Project Management Institute (2017). *A guide to the Project Management Body of Knowledge (PMBOK guide)*. 6. Newton Square, PA: Project Management Institute.
- (2021). *A Guide to the Project Management Body of Knowledge: PMBOK (®) Guide*. Project Management Institute.
- Qahtani, Abdulrahman M. (2020). “An Empirical Study of Agile Testing in A Distributed Software Development Project”. Em: *Proceedings of the 2020 3rd International Conference on Geoinformatics and Data Analysis*. ICGDA 2020. Marseille, France: Association for Computing Machinery, pp. 110–114. ISBN: 9781450377416. DOI: [10.1145/3397056.3397085](https://doi.org/10.1145/3397056.3397085). URL: <https://doi.org/10.1145/3397056.3397085>.
- Ramirez-Mora, Sandra L e Hanna Oktaba (2017). “Productivity in agile software development: a systematic mapping study”. Em: *2017 5th international conference in software engineering research and innovation (CONISOFT)*. IEEE, pp. 44–53.
- Santos, Ronnie E. S. (2017). *Uses, Benefits, and Limitations of Job Rotation in Software Engineering*. DOI: [10.1145/3011286.3011300](https://doi.org/10.1145/3011286.3011300).
- Santos, Ronnie E. S., F. Silva e C. V. C. Magalhães (2016). *Benefits and limitations of job rotation in software organizations: a systematic literature review*. DOI: [10.1145/2915970.2915988](https://doi.org/10.1145/2915970.2915988).
- Santos, Ronnie E. S., Fabio Q. B. da Silva et al. (2016). *Building a theory of job rotation in software engineering from an instrumental case study*. DOI: [10.1145/2884781.2884837](https://doi.org/10.1145/2884781.2884837).
- Santos, Ronnie ES (2017). “Uses, Benefits, and Limitations of Job Rotation in Software Engineering”. Em: *ACM SIGSOFT Software Engineering Notes* 41.6, pp. 1–6.
- Schloegel, Uta et al. (2016). “Reducing age stereotypes in software development: The effects of awareness- and cooperation-based diversity interventions”. Em: *Journal of Systems and Software* 121, pp. 1–15. ISSN: 0164-1212. DOI: <https://doi.org/10.1016/j.jss.2016.07.041>. URL: <https://www.sciencedirect.com/science/article/pii/S0164121216301352>.
- Scott, Ezequiel, Khaled Nimr Charkie e Dietmar Pfahl (2020). “Productivity, turnover, and team stability of agile teams in open-source software projects”. Em: *2020 46th Euro-micro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, pp. 124–131.

- Shahzad, Fakhar, GuoYi Xiu e Muhammad Shahbaz (2017). “Organizational culture and innovation performance in Pakistan’s software industry”. Em: *Technology in Society* 51, pp. 66–73. ISSN: 0160-791X. DOI: <https://doi.org/10.1016/j.techsoc.2017.08.002>. URL: <https://www.sciencedirect.com/science/article/pii/S0160791X17300787>.
- Shameem, Mohammad et al. (2020). “Taxonomical classification of barriers for scaling agile methods in global software development environment using fuzzy analytic hierarchy process”. Em: *Applied Soft Computing* 90, p. 106122. ISSN: 1568-4946. DOI: <https://doi.org/10.1016/j.asoc.2020.106122>. URL: <https://www.sciencedirect.com/science/article/pii/S1568494620300624>.
- Shen, Xiao-Ning et al. (2018). “A Q-learning-based memetic algorithm for multi-objective dynamic software project scheduling”. Em: *Information Sciences* 428, pp. 1–29. ISSN: 0020-0255. DOI: <https://doi.org/10.1016/j.ins.2017.10.041>. URL: <https://www.sciencedirect.com/science/article/pii/S0020025517310472>.
- Shen, Xiaoning, Yinan Guo e Aimin Li (2020). “Cooperative coevolution with an improved resource allocation for large-scale multi-objective software project scheduling”. Em: *Applied Soft Computing* 88, p. 106059. ISSN: 1568-4946. DOI: <https://doi.org/10.1016/j.asoc.2019.106059>. URL: <https://www.sciencedirect.com/science/article/pii/S1568494619308415>.
- Sommerville, Ian (2010). “Software testing”. Em: *Software Engineering* 8, pp. 537–65.
- Song, Xuhong et al. (2020). “Optimization of Human Resource Allocation Problem Considering Costs Reduction and Balance in Chinese State-Owned Enterprise”. Em: *Proceedings of the 2020 4th International Conference on Management Engineering, Software Engineering and Service Sciences*. ICMSS 2020. Wuhan, China: Association for Computing Machinery, pp. 211–215. ISBN: 9781450376419. DOI: [10.1145/3380625.3380658](https://doi.org/10.1145/3380625.3380658). URL: <https://doi.org/10.1145/3380625.3380658>.
- Stylianou, Constantinos e Andreas S. Andreou (2016). “Investigating the impact of developer productivity, task interdependence type and communication overhead in a multi-objective optimization approach for software project planning”. Em: *Advances in Engineering Software* 98, pp. 79–96. ISSN: 0965-9978. DOI: <https://doi.org/10.1016/j.aes.2016.05.002>.

- 1016/j.advengsoft.2016.04.001. URL: <https://www.sciencedirect.com/science/article/pii/S096599781630076X>.
- Tam, Carlos et al. (2020). “The factors influencing the success of on-going agile software development projects”. Em: *International Journal of Project Management* 38.3, pp. 165–176.
- Telemaco, Ulisses et al. (2020). “A catalogue of agile smells for agility assessment”. Em: *IEEE Access* 8, pp. 79239–79259.
- Tighy, Gert (2012). “Evaluation of software engineering management best practices in the Western Cape”. Em: *Proceedings of the 2012 4th IEEE Software Engineering Colloquium (SE)*, pp. 21–23. DOI: [10.1109/SE.2012.6242352](https://doi.org/10.1109/SE.2012.6242352).
- Trinkenreich, Bianca et al. (2023). “A Model for Understanding and Reducing Developer Burnout”. Em: *arXiv preprint arXiv:2301.09103*.
- Vega-Velazquez, Miguel Ángel, Abel Garcia-Najera e Humberto Cervantes (2018). “A survey on the Software Project Scheduling Problem”. Em: *International Journal of Production Economics* 202, pp. 145–161. ISSN: 0925-5273. DOI: <https://doi.org/10.1016/j.ijpe.2018.04.020>. URL: <https://www.sciencedirect.com/science/article/pii/S0925527318301828>.
- Vishnubhotla, Sai Datta, Emilia Mendes e Lars Lundberg (2020). “Investigating the relationship between personalities and agile team climate of software professionals in a telecom company”. Em: *Information Software Technology*. DOI: [10.1016/J.INFSOF.2020.106335](https://doi.org/10.1016/J.INFSOF.2020.106335).
- von Wangenheim, Christiane Gresse et al. (2010). “Best practice fusion of CMMI-DEV v1.2 (PP, PMC, SAM) and PMBOK 2008”. Em: *Information and Software Technology* 52.7, pp. 749–757. ISSN: 0950-5849. DOI: <https://doi.org/10.1016/j.infsof.2010.03.008>. URL: <https://www.sciencedirect.com/science/article/pii/S0950584910000480>.
- Wang, Lu, Minxue Huang e Maohong Liu (2018). “How founders’ social capital affects the success of open-source projects: A resource-based view of project teams”. Em: *Electronic Commerce Research and Applications* 30, pp. 51–61. ISSN: 1567-4223. DOI: <https://doi.org/10.1016/j.elerap.2018.05.003>. URL: <https://www.sciencedirect.com/science/article/pii/S1567422318300437>.

- Warrick, D. (2017). “What leaders need to know about organizational culture”. Em: *Business Horizons* 60, pp. 395–404. DOI: [10.1016/J.BUSHOR.2017.01.011](https://doi.org/10.1016/J.BUSHOR.2017.01.011).
- Xia, Xin, D. Lo et al. (2017). *Personality and Project Success: Insights from a Large-Scale Study with Professionals*. DOI: [10.1109/ICSME.2017.50](https://doi.org/10.1109/ICSME.2017.50).
- Xia, Xin, David Lo et al. (2017). “Personality and Project Success: Insights from a Large-Scale Study with Professionals”. Em: *2017 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 318–328. DOI: [10.1109/ICSME.2017.50](https://doi.org/10.1109/ICSME.2017.50).
- Yilmaz, Murat et al. (2017). “An examination of personality traits and how they impact on software development teams”. Em: *Information Software Technology*. DOI: [10.1016/J.INFSOF.2017.01.005](https://doi.org/10.1016/J.INFSOF.2017.01.005).
- Zaouga, Wiem, Latifa Ben Arfa Rabai e Wafa Rashid Alalyani (2019). “Towards an ontology based-approach for human resource management”. Em: *Procedia Computer Science* 151, pp. 417–424.
- Zapotecas-Martinez, Saúl, Abel Garcia-Nájera e Humberto Cervantes (2020). “Multi-objective optimization in the agile software project scheduling using decomposition”. Em: *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, pp. 1495–1502.