

REESTRUTURAÇÃO DE UM *SQUAD* DE DESENVOLVIMENTO DE *SOFTWARE* ÁGIL EM DOIS NOVOS *SQUADS*: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA

Andressa Cristina de Almeida Butafava; Universidade Nove de Julho; UNINOVE
andressabutafava@hotmail.com

Luciano Ferreira da Silva; Universidade Nove de Julho; UNINOVE
prof.lfs7725@gmail.com

Renato Penha; Universidade Nove de Julho; UNINOVE
rp.renatopenha@gmail.com

RESUMO

Objetivo: o presente estudo busca entender quais as consequências geradas na performance de uma equipe de desenvolvimento de *software* a partir de sua reestruturação em dois *Squads* em um ambiente de abordagem ágil em uma instituição financeira brasileira. Entende-se a relevância do tema, visto a crescente demanda mundial de desenvolvimento de *software* e aumento do uso de abordagem ágil.

Método: a pesquisa utilizou uma abordagem qualitativa, por meio de um estudo de caso único. O processo de coleta de dados se deu por meio de doze entrevistas semiestruturadas e por análise de documentos. Depois dos dados coletados foi aplicada análise de conteúdo nos dados textuais e uso de estatística descritiva para os registros das atividades.

Resultados: após a reestruturação da equipe em dois *Squads*, os indicadores mostraram que os times tiveram melhora na performance por interferência de vários fatores como: ganho de benefícios, aumento de competências e uso adequado de ferramentas e práticas de gerenciamento de projetos.

Contribuição: a pesquisa levantou fatores que podem ser explorados se houver necessidade de reestruturar um *squad*, sendo eles: preparar e disponibilizar todos papéis dos membros que comporão cada *Squads*; comunicar os membros com transparência a respeito do processo da quebra, incluindo os fatores motivadores e o método utilizado para divisão dos times; treinar os novos integrantes que assumirão papéis diferentes do que desempenhavam antes; e alocar os projetos no *Squads* correto de acordo com seu escopo.

Palavras-chave: Gerenciamento de Projetos; Abordagem ágil; Desenvolvimento de *software*; Competências; Instituições financeiras.

Data de recebimento: 12/08/2022

Data do aceite de publicação: 04/11/2022

Data da publicação: 30/12/2022

RESTRUCTURING AN AGILE SOFTWARE DEVELOPMENT SQUAD INTO TWO NEW SQUADS: A CASE STUDY IN A FINANCIAL INSTITUTION

ABSTRACT

Objective: this technical report seeks to understand the consequences in the performance of a software development team break into two different squads in an agile approach environment in a Brazilian financial institution. The relevance of the topic is given by the growing demand for software development and the increase in the use of agile approach.

Method: this research used a qualitative approach, through a study case. The data collection process took place through twelve semi-structured interviews and document analysis. The data were analyzed through the textual data and using descriptive statistics for activity records.

Results: after the restructuring of the team in two squads, the performance indicators showed that bought teams had a better performance through interference of several factors such as: gain of benefits, increase in skills and proper use of project management tools and practices.

Contribution: the research raised some factors that can be better explored if there is a need to restructure a squad into two new ones: prepare and make available all the functions of the members that will compose each squad; communicate members about the breakdown process, including the factors that motivated it and the method used to break the teams; empower new members who will assume different positions than they played before; and allocate the projects in the correct squad according to their scope.

Keywords: Project Management; Agile approach; Software development; Competences; Financial Institutions.

1 INTRODUÇÃO

O setor de Tecnologia da Informação (TI) é um dos segmentos mais dinâmicos de toda economia, caracterizado pela crescente inovação, pequeno ciclo de vida dos produtos, tecnologia, prestação de serviços, juntamente com a rápida difusão de inovações em vários mercados estrangeiros (Cizmas *et al.*, 2020). Neste contexto, a habilidade das companhias de gerar inovações tem um impacto direto no negócio em termos de criação de vantagem competitiva, que pode ajudar as companhias a garantir seu desenvolvimento sustentável em longo prazo (Paletta & Vieira, 2008).

Embora diversos recursos contribuam para sucesso das companhias, parte da solução está na melhoria da equipe de trabalho, porque a singularidade do desenvolvimento de *software* é amplamente baseada no esforço da equipe como apontam Yusoff *et al.* (2012). Conseqüentemente, esse desenvolvimento é compreendido por um processo criativo que pode envolver muita interação humana no contexto de time. Por isso, é importante entender os fatores do time que podem influenciar em sua performance (Hsu *et al.*, 2012).

Ainda no sentido da capacidade de inovar e as equipes de desenvolvimento de *software*, acredita-se que, para além de modelos inovadores e práticas consolidadas de gerenciamento de

projetos, a alocação de recursos por competências e habilidades seja fundamental para potencializar a produtividade e o valor entregue pelas empresas (Gonçalves *et al.*, 2020). Deste modo, apesar de serem comuns as atividades em conjunto nas organizações, se faz necessário compreender diferenças neste tipo de trabalho e as especificidades relacionadas às pessoas que desempenham algum papel para o alcance dos objetivos.

Levi e Slem (1995) definem um grupo por um conjunto de pessoas que trabalham juntas para performar uma tarefa. Nele, os membros dividem um objetivo em comum e são coordenados por um líder, mas sua avaliação de performance é individual. Em contraste, Katzenbach e Smith (1993) descrevem uma equipe como sendo composta por um pequeno número de pessoas com habilidades complementares que estão comprometidas com um propósito comum. Logo, uma equipe é representada por ter um conjunto de objetivos e adotar uma visão pela qual todos os membros se consideram mutuamente responsáveis.

Portanto, pode-se dizer que há uma diferença considerável entre estes dois tipos de trabalho em conjunto. Nesse contexto, as empresas têm implantado novos conceitos para composição de times e atribuição de papéis aos colaboradores. Um deles é o *squad*, também conhecido por “esquadrões”, sendo um time multifuncional que ajuda a alcançar uma solução criativa e inovadora (Ganis & Waszkiewicz, 2018). Um *squad* é projetado para dar maior velocidade de desenvolvimento aos produtos por meio de entregas parciais e prestação de serviços da empresa, uma vez que contribui para reduzir o desperdício e minimizar o tempo de inatividade da equipe (Gonçalves *et al.*, 2020).

Para tratar do contexto que este estudo aborda é preciso descrever o ambiente das instituições financeiras. Assim, com o aumento da inovação tecnológica e da concorrência entre as instituições financeiras e *fintechs*, percebe-se que as competências dos times de projetos de desenvolvimento de *software* colaboram fortemente para que eles obtenham uma melhor performance (Jiang, 2009; Doherty, Ashurst & Peppard, 2012).

A literatura fornece informações sobre a estrutura dos *squads*. Segundo Gonçalves *et al.* (2020) é indicado que o *squad* seja formado por um grupo de três a dez membros. Para Ganis e Waszkiewicz (2018), seguindo as diretrizes de gerenciamento ágil de projetos, é indicado que o tamanho de uma equipe não exceda nove membros. Porém, pode acontecer que esse time acabe crescendo e em algum momento, para melhoria de sua performance, seja necessário reestruturá-lo.

As seções deste estudo seguem as oito etapas contidas no artigo de Biancolino *et al.* (2012), dividindo-se em: Introdução, Contexto do Projeto ou Situação-Problema, Referencial Teórico, Método da Produção Técnica, Tipo de Intervenção e Mecanismos Adotados, Resultados Obtidos e Análise, Considerações Finais e Referências.

Inicialmente, o *squad* tema desse estudo era composto por dezessete integrantes, sendo onze colaboradores da instituição financeira e seis terceirizados. Na área de tecnologia da informação da empresa estudada, o *squad* é composto pela tríade (*scrum master*, *product owner* e líder técnico), além dos desenvolvedores de *software*. Estes podem ser funcionários da instituição ou terceirizados. No caso desse *squad* em estudo, todos os terceirizados faziam o papel de desenvolvedores de *software*.

O time estava com dificuldade de gerenciar as tarefas dos projetos em execução e a comunicação entre seus membros, pois várias demandas eram executadas ao mesmo tempo. Então, criaram-se dentro desse *squad* pequenos grupos que cuidavam de temas diferentes e não se comunicavam no dia a dia. Além disso, durante as cerimônias, o time não conseguia ter a compreensão por completo de tudo o que ali era executado. Também era difícil que os membros conseguissem se especializar em um único assunto.

Neste contexto, se faz necessário compreender que o gerenciamento de projetos por meio de métodos ágeis tem por objetivo tornar o processo de gestão mais simples, flexível e

prático (Amaral *et al.*, 2011). Dessa forma, levando em conta que a realidade de gestão evidenciada estava alinhada aos pressupostos dos métodos ágeis, foi proposta a quebra dessa equipe em dois *squads* diferentes de modo a alavancar a performance da equipe e conseguir, por meio da especialização de seus integrantes em assuntos específicos, entregar uma maior quantidade de projetos com maior qualidade e em um menor período.

Com base no que foi apresentado sobre o desenvolvimento de *softwares* em equipes no contexto de uma instituição financeira, este estudo tem como objetivo responder a pergunta de pesquisa: Quais as consequências na performance decorrentes da decomposição de uma equipe de desenvolvimento de software usando abordagem ágil em duas equipes em uma instituição financeira brasileira?

2 REFERENCIAL TEÓRICO

O gerenciamento de projetos passou a ser visto como uma atividade estratégica com o objetivo de criar uma vantagem competitiva para as organizações (Patanakul & Shenhar, 2012). De acordo com o PMI (2021), o gerenciamento de projetos cria valor de diferentes formas: por meio da criação de um novo produto, serviço ou resultado que atenda aos desejos do usuário final; construindo contribuições positivas, sociais ou ambientais; aprimorando a eficiência, produtividade, eficácia ou a capacidade de resposta; colaborando para a transição organizacional; e colaborando para que os benefícios criados por programas, projetos ou operações sejam sustentados.

Gerenciar um projeto envolve, entre outras atividades, realizar o balanceamento das restrições conflitantes do projeto que incluem, mas não se limitam a: escopo, qualidade, cronograma, orçamento, recursos e risco (de Medeiros, 2011). Um projeto é “um esforço temporário empreendido para criar um produto, serviço ou resultado único” (PMI, 2021 p. 251).

De acordo com Almeida, Campagnac e Farias Filho (2008), o gerenciamento de projetos está inserido dentro das organizações como uma prática crescente e essencial, com inúmeros benefícios e de uma importância vital para os negócios. Nesse contexto, é preciso compreender como se dá a formação de equipes como um processo que agrega um conjunto de indivíduos com diferentes necessidades, habilidades e inteligências (Thamhain, 1993).

Ahmad *et al.* (2016) explicam que muitas corporações migraram para o uso de times de projeto para investigar, criar e dispensar itens e serviços ou examinar questões, especialmente em tarefas complicadas, a fim de melhorar a performance das companhias. Não obstante a relevância das capacidades individuais, o benefício na gestão de equipes é transformá-las em uma unidade de trabalho eficaz e integrada que promova melhores resultados (Levi & Slem, 1995).

2.1 Performance do time

A performance do time pode ser definida como uma medida em que a equipe é capaz de atender aos objetivos estabelecidos de qualidade, custo e tempo (Lechler, 1997). Fung e Ali (2011) descobriram que a performance do time é sempre mais efetiva que a performance individual. Assim, as organizações tomam um grande cuidado em formar a combinação correta de indivíduos para criar os melhores times (McClough & Rogelberg, 2003), e assim obter os melhores resultados.

Nesse contexto, a equipe atua como um dos recursos limitados e mais significativos de um projeto (Sudhakar, 2010). Além disso, vale destacar que a performance do time depende de uma série de fatores como: a tarefa confiada à equipe, os recursos disponíveis, o contexto organizacional em que os times estão envolvidos, a diversidade de membros, os níveis

diferenciados de conhecimento, os processos operacionais da equipe, entre outros (Tirmizi, 2008).

Por muitas décadas, aspectos da performance do time, incluindo seus processos e os efeitos nos resultados do trabalho foram estudados. Lu *et al.* (2006) destacam que um aspecto relevante neste contexto é a virtualização do trabalho que pode ser entendida como a condição na qual os times virtuais trabalham. Times virtuais são formados por trabalhadores dispersos geograficamente, mas reunidos por meio de tecnologias de informação e comunicação (Piccoli & Ives, 2004). Na medida em que as equipes se tornam cada vez mais virtuais, seu desempenho se torna correspondentemente mais complexo (Lu *et al.*, 2006). A literatura sugere quatro grandes áreas de trabalho em que as equipes são impactadas pela virtualidade: comunicação, confiança, participação e coordenação do time e resultados do time (Lu *et al.*, 2006).

Em resumo, a análise dos resultados de uma equipe está baseada na sua capacidade de trabalhar de forma integrada (Levi & Slem 1995). Neste sentido, Richards (2003) reconheceu que o conhecimento individual de um integrante da equipe e sua expertise não são suficientes para o time obter os resultados esperados. Portanto, todo projeto de *software* vai inevitavelmente encarar a questão da composição da equipe e os pesquisadores perceberam que a diversidade nesse momento promoveria sucesso à equipe, ajudando um projeto a conquistar sua missão, visão e valores (Liang & Liu, 2007).

Para quantificar o sucesso da performance da equipe, pesquisadores tem estudado características de personalidade, relações interpessoais e interações entre os membros do time (Barki & Hartwick, 2001). Neste contexto, Ganis e Waszkiewicz (2018) explicam que a comunicação é percebida como um fator importante para o sucesso do projeto. Uma comunicação ruim pode causar atrasos, falta de entendimento, erros, confusão e falhas (Sudhakar, 2012).

2.2 Competências

A competência é a habilidade de um indivíduo, ou grupo, de agir e reagir com a relevância requerida para desempenhar um conjunto de atividades em situações diversas para determinado tipo de trabalho (Sefiani et al., 2012). Rabechini et al. (2011) explicam que a competência é um conceito pelo qual se definem quais são as atitudes, as habilidades e os conhecimentos necessários para alcançar resultados diferenciados.

Fleury e Fleury (2001) descrevem o modelo de competências a partir da combinação das dimensões reconhecidas como CHA, sendo elas conhecimento, habilidade e atitude. Estas dimensões da competência se materializam no saber por meio do conhecimento adquirido, no saber fazer com base na aplicação das habilidades e na disposição para aplicar conhecimentos e habilidades com base na atitude.

A competência comunicação utilizada no desenvolvimento de *software*, é frequentemente estudada como um mecanismo para os membros da equipe trocarem informações, resolverem a ambiguidade e a incerteza decorrentes de tarefas complexas e inter-relacionadas (Brodbeck, 2001). Para melhor compreender os parâmetros em que são analisadas as competências de uma equipe, é preciso observar alguns cenários em que cada equipe opera, a abordagem de projetos e as técnicas utilizadas (Gonçalves *et al.*, 2020).

Com base neste contexto, pode-se dizer que as competências múltiplas e específicas de uma equipe de profissionais podem ser fundamentais para que os projetos alcancem os melhores resultados para suas organizações. Hoegl e Gemuenden (2001) investigaram os seis efeitos que influenciam a qualidade do trabalho, sendo eles a coesão, balanço das contribuições dos membros, comunicação, suporte mútuo, esforço e coordenação.

Gonçalves *et al.* (2020) identificaram quais competências têm sido demandadas dos profissionais que atuam em projetos, no âmbito de abordagens ágeis, a fim de orientar gestores

de projetos a formar equipes de alta performance. Os resultados obtidos mostram que as competências encontradas foram categorizadas em três dimensões:

Conhecimento: é composta pelas categorias conhecimento técnico, conhecimento da abordagem de projetos e conhecimento das ferramentas de acompanhamento e comunicação;

Habilidade: compreende as categorias comunicação, saber ouvir, flexibilidade, saber trabalhar em equipe, liderança e capacidade de execução;

Atitude: engloba as categorias aderir às cerimônias do scrum, colaboração, adaptabilidade e proatividade.

2.3 *Desenvolvimento de software e gerenciamento de projetos ágeis*

O desenvolvimento de *software* é em grande parte uma atividade humana, sendo intensivo em mão de obra e conhecimento, e as organizações desse mercado operam em ambientes altamente voláteis nos quais diferentes elementos de desempenho podem mudar rapidamente (Fagerholm *et al.*, 2015).

Por abordagens ágeis de gerenciamento de projetos entende-se a onda de práticas de gerenciamento que se seguiu aos modelos tradicionais, como alternativa para um ambiente de negócios que se apresenta cada vez mais dinâmico, incerto e adaptativo, em que os projetos passam constantemente por mudanças de requisitos e que os resultados precisam ser entregues ao cliente em pequenos intervalos de tempo (Sutherland & Schwaber, 2013). A adoção de métodos ágeis no desenvolvimento ágil de software tem experimentado um enorme crescimento e é amplamente reconhecida na prática (Palopak *et al.*, 2022).

As técnicas ágeis são focadas em serem mais flexíveis e iterativas na abordagem, resultando na redução do risco, promovendo qualidade, adaptabilidade e maior satisfação do cliente (Crispin & Gregory, 2009). Técnicas de desenvolvimento ágil têm sido empregadas para melhorar o desempenho do projeto (Moe, Cruzes, Dybå, & Engebretsen, 2015). Ainda sobre a abordagem ágil, ela é caracterizada pelo trabalho colaborativo, requerindo habilidades multidisciplinares, tomada de decisão pluralista, alto envolvimento do cliente e equipes pequenas (Lindsjörn *et al.*, 2016).

Comparado a outros métodos de desenvolvimento de sistemas, o ágil tem sido bem recebido pelos profissionais, pois experimentou o aumento da satisfação do cliente, reduziu as taxas de falhas, reduziu a duração do desenvolvimento e responde rapidamente às mudanças nos requisitos (Dybå *et al.*, 2008).

Neste contexto, se faz necessário destacar que com a crescente taxa de adoção de desenvolvimento ágil de *software* em várias organizações nas últimas décadas, a necessidade do trabalho colaborativo aumentou, exigindo habilidades multidisciplinares e habilidades interpessoais dos membros da equipe (Lindsjörn *et al.*, 2016). O ágil se tornou uma disciplina crucial de engenharia de software tanto na prática quanto na pesquisa (Hoda *et al.*, 2018). Sua popularidade se deve à sua simplicidade, adaptabilidade e estabilidade para as demandas atuais de desenvolvimento de software (Yang *et al.*, 2016).

No método ágil mais popular, *Scrum*, o trabalho é organizado em times pequenos e multifuncionais com um facilitador e *team members* (Lindsjörn *et al.*, 2016). Esses *team members* coordenam seu trabalho frequentemente nas reuniões chamadas de *daily stand-up meetings* (Stray *et al.*, 2016). Sharp e Robinson (2010) descreveram como as equipes de desenvolvimento ágil colaboram, coordenam e se comunicam. Pikkarainen *et al.* (2008) focaram seus estudos em como os métodos de desenvolvimento ágil podem melhorar a comunicação, sendo que os autores afirmam que as práticas *Scrum* melhoram a comunicação formal e informal. Sobre os fatores que podem tornar mais produtiva uma equipe que trabalha com abordagem ágil, podemos destacar: experiência do time, competências, motivação e comunicação (Melo *et al.*, 2011).

2.4 Medidas de desempenho

Com a expansão do setor da Tecnologia da Informação e a necessidade de as companhias acompanharem seu desempenho organizacional, os indicadores de gerenciamento podem ser utilizados como forma de avaliar o sucesso de um projeto (Grander *et al.*, 2019). Nesse sentido, projetos bem-sucedidos entregam os benefícios esperados e em seguida criam valor estratégico para os negócios (Serra & Kunc, 2015). Os indicadores de desempenho nos processos de negócios são valores-alvo que devem ser alcançados em um determinado período de análise (del-Rio-Ortega *et al.*, 2013).

Em nível de gerenciamento de projetos, todas as empresas de software precisam medir a produtividade de seus projetos, pois isso permite obter indicadores para gerenciar e avaliar processos, projetos, produtos e pessoas (Machuca-Villegas *et al.*, 2021). Os resultados dessas medições são usados para tomar decisões e melhorar seus projetos de software. Nesse contexto, os principais objetivos do uso de métricas para o processo de desenvolvimento de software são: (i) alinhar os objetivos de negócios com TI, (ii) ajudar a alcançar os requisitos de conformidade para operações de negócios e (iii) impulsionar a excelência operacional dos serviços de TI (Zheng *et al.*, 2021).

Por conta desses fatores, é importante considerar a produtividade da equipe de projeto de desenvolvimento de software, uma vez que software “é um produto direto dos processos cognitivos de indivíduos engajados em trabalho em equipe inovador e intensivo em intelecto” (PMI, 2013). Portanto, o estudo das métricas e seu efeito na produtividade do desenvolvimento de software é de fato um assunto de especial interesse para as empresas de software (de Barros *et al.*, 2010). A adoção de práticas ágeis é essencial para melhorar o desempenho das métricas nos projetos de software. Muitas empresas que implementaram os métodos ágeis mostram que eles impactam positivamente o desempenho do projeto de desenvolvimento de software (Palopak *et al.*, 2022).

3 METODOLOGIA

Essa pesquisa utilizou uma abordagem qualitativa, por meio de um estudo de caso único. Esse tipo de investigação requer a realização de uma coleta de dados que se dá por meio de observação direta, entrevistas, documentos e registros em arquivos (Bryman, 1989). A primeira etapa da pesquisa consistiu em buscar respaldo teórico para proporcionar fundamentos ao estudo e obter um maior aprofundamento sobre o tema. Este estudo contemplou os seguintes assuntos: gerenciamento de projetos ágeis, instituições financeiras, competências e equipes de desenvolvimento de *software*.

O estudo foi realizado em uma instituição financeira brasileira de grande porte, uma empresa com forte cultura de inovação e que desenvolve produtos financeiros. A segunda etapa consistiu em realizar entrevistas, orientadas por um instrumento de coleta de dados semiestruturado. Importante ressaltar que a área de Tecnologia da Informação da empresa em estudo adotou o modelo virtual como forma de trabalho desde março/2020.

As entrevistas foram realizadas de forma virtual com doze colaboradores da empresa que participaram da equipe pré e pós reestruturação em dois *squads*. As entrevistas foram gravadas, totalizando 185 minutos de gravação. As entrevistas também foram transcritas com o auxílio do *software Microsoft Word* (2010) totalizando 31 páginas.

No *squad* 1, responsável pela contratação do produto cheque especial, quatro colaboradores foram entrevistados, sendo: E1, E4, E5 e E10. No *squad* 2, responsável pela manutenção do cheque especial, também foram entrevistados outros quatro colaboradores: E3,

REESTRUTURAÇÃO DE UM *SQUAD* DE DESENVOLVIMENTO DE *SOFTWARE* ÁGIL EM DOIS NOVOS *SQUADS*: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA

E6, E11 e E12. Os demais entrevistados exercem papel de liderança na empresa: E2, E7, E8 e E9. A Tabela 1 mostra o perfil dos participantes das entrevistas.

Tabela 1 – Perfil dos participantes das entrevistas (em anos)

	Papel	Idade	Tempo de experiência no mercado	Tempo de trabalho na instituição financeira
Entrevistado 1	<i>Product Owner</i>	29	10	1
Entrevistado 2	Coordenador dos desenvolvedores de <i>software</i> e do líder técnico	38	20	10
Entrevistado 3	<i>Product Owner</i>	38	19	4
Entrevistado 4	Desenvolvedor de <i>software</i>	26	6	2
Entrevistado 5	Desenvolvedor de <i>software</i>	28	10	5
Entrevistado 6	Desenvolvedor de <i>software</i>	60	40	1
Entrevistado 7	Coordenadora dos <i>Product Owners</i>	47	29	25
Entrevistado 8	Coordenador dos <i>Scrum masters</i>	41	21	19
Entrevistado 9	Gerente dos desenvolvedores de <i>software</i>	59	40	35
Entrevistado 10	Desenvolvedor de <i>software</i>	43	28	21
Entrevistado 11	Desenvolvedor de <i>software</i>	28	12	3
Entrevistado 12	Desenvolvedor de <i>software</i>	26	6	4

Fonte: elaborado pelos autores.

A média de idade dos entrevistados é de 39 anos, o tempo médio de experiência no mercado é de 20 anos e o tempo de trabalho na instituição financeira em análise é de 11 anos. Cada entrevistado foi escolhido com base em sua habilidade de transmitir o ocorrido na situação de reestruturação do *squad*. A coleta de dados a partir de entrevistas é adequada para entender a situação real do entrevistado (de Moraes *et al.*, 2020; Russo & da Silva, 2019).

Em seguida, as entrevistas foram transcritas para facilitar a compreensão. No processo de análise foi utilizada a técnica de codificação baseada nas prescrições de Charmaz (2006). Essa técnica consiste na segmentação do texto em códigos. O *software* Atlas.Ti (2012) foi utilizado para apoiar essa atividade (Friese, 2012). Vale ressaltar que no *software* é possível criar códigos e atribuí-los ao texto.

Assim, inicialmente foi utilizada uma codificação aberta, identificando as categorias que seriam úteis para atingir o objetivo dessa pesquisa. Depois foi realizada uma codificação axial, procurando uma relação de causa e efeito entre os códigos, entrevistas e a teoria que apoia esse estudo. O *software* utilizado também permite relacionar os códigos aos trechos das entrevistas, assim como há a opção de gerar uma tabela contendo a quantidade que os códigos que foram mencionados. Com relação ao processo de codificação, é necessário explicitar que alguns códigos podem ser atribuídos no mesmo trecho da entrevista. Quando acontece essa sobreposição, ou mesmo aproximação dos códigos, o *software* entende a situação como uma co-ocorrência.

A terceira etapa da pesquisa consistiu em coletar e analisar os indicadores pelos quais os *squads* são monitorados, sendo eles: produtividade, *leadtime* de *story*, *throughput* de BCP (*Business Complexity Points*), *throughput* de *story*, quantidade de mudanças e quantidade de falhas nas mudanças.

Por fim, realizou-se a análise dos resultados, verificando as consequências geradas na performance da equipe a partir de sua quebra em dois *squads* diferentes.

4 ANÁLISE DOS DADOS

Visando melhorar a performance dessa equipe, foi definido pela liderança que o *squad* 1 cuidaria dos assuntos relacionados à contratação do cheque especial. Já o *squad* 2, trataria dos assuntos relacionados à manutenção do mesmo produto conforme apresentado na Figura 1.

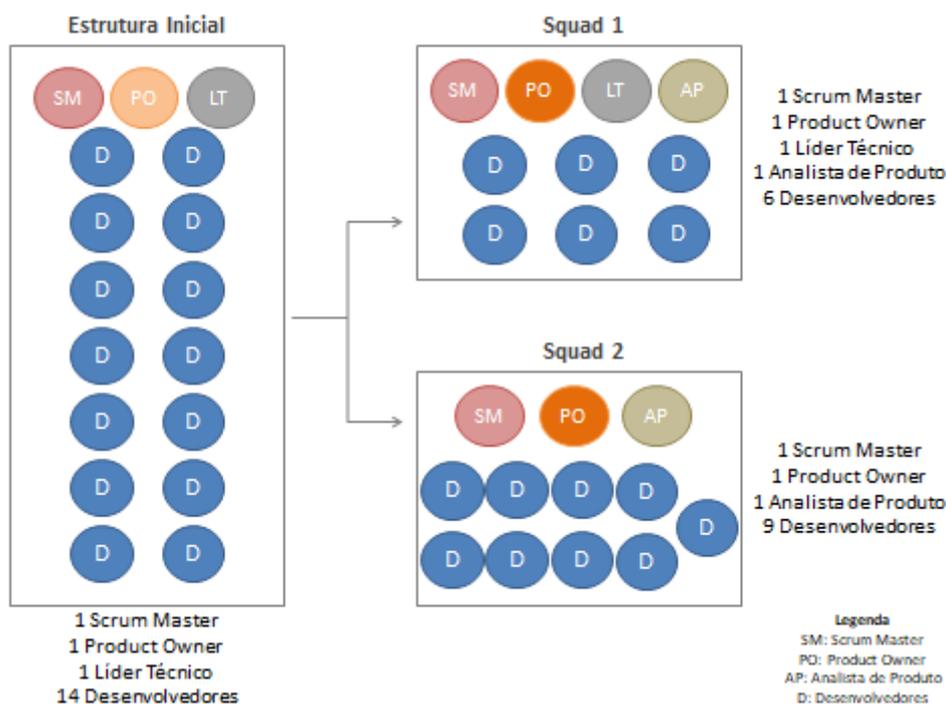


Figura 1 – Configuração da equipe antes e depois da divisão em dois *squads*
Fonte: elaborado pelos autores.

Portanto, a intervenção proposta foi de divisão nas citadas *squads*, com o objetivo de melhorar o desempenho dos projetos, com entregas mais rápidas, possibilidade de testes e inovações ao ter menor custo de gestão de mudanças, enxugar os processos e entrega de maior valor pelas empresas (Gonçalves *et al.*, 2020).

Além da separação da equipe, também foi definido que seriam introduzidos novos papéis nessas equipes. No *squad* 1, um analista de produto foi alocado. Este papel tem o objetivo de compartilhar o conhecimento sobre o produto com a equipe, agilizar a comunicação entre a área de tecnologia da informação com negócios e colaborar na priorização do *backlog* do produto.

No *squad* 2 também foi incluído um analista de produto e um *product owner*. Este papel teria as mesmas funções que um *product owner* desenvolve, mas com a vantagem de um grande conhecimento da área de negócios sobre o produto do *squad*. Cabe ressaltar que nos primeiros três meses após a divisão das equipes, os *squads* 1 e 2 ficaram com o líder técnico dividido entre elas, mas esse papel estava oficialmente alocado no *squad* 1.

Para melhor entender o resultado da intervenção foram utilizados algumas métricas relacionados aos períodos antes e depois da reestruturação da equipe em dois *squads*.

A produtividade em desenvolvimento de software pode ser definida como a razão entre o *output* e o *input* dentro do seu processo de desenvolvimento (Machuca-Villegas *et al.*, 2021). O *output* é o resultado gerado, podendo ser o serviço final, as tarefas executadas, linhas de código, mudanças implementadas, entre outros. O *input* é o esforço dedicado para alcançar o *output* (tempo, custo de mão de obra, número de recursos e outros). Nesse estudo, a

REESTRUTURAÇÃO DE UM *SQUAD* DE DESENVOLVIMENTO DE *SOFTWARE* ÁGIL EM DOIS NOVOS *SQUADS*: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA

produtividade é medida pela quantidade de horas necessárias para se ativar um BCP (*Business Complexity Point*), ou seja, quanto menor, melhor. Os BCPs são pontos normalizados para estimativa ou definição da complexidade funcional do desenvolvimento de software, possibilitando a mensuração do tamanho do software a ser construído (Data Driven Agile, 2015). Para o período em estudo, a produtividade dos *squads* 1 e 2 melhorou em relação ao *squad* único.

O *throughput* é a habilidade de entregar o máximo possível por unidade em um período de tempo e é mensurado pelo número de itens entregues (Mannila, 2013). O *throughput* de BCP mede a quantidade de *business complexity points* ativados no período de um mês. Para esse indicador, quanto mais pontos ativados, melhor o resultado da equipe. Após a reestruturação do *squad* é possível perceber que houve uma melhoria nesse indicador. O comportamento desses dois indicadores pode ser observado na Figura 2.

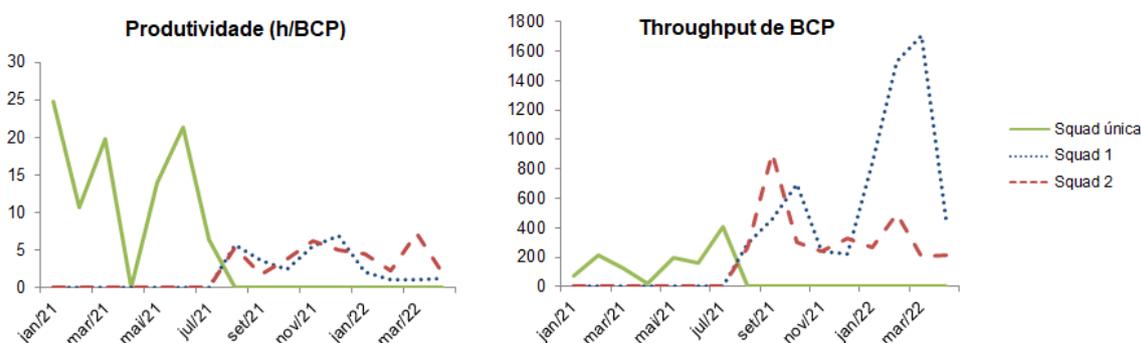


Figura 2 – Indicadores de produtividade e *throughput* de BCP

Fonte: elaborado pelos autores.

Uma *story* é composta por uma ou mais sentenças que descrevem as necessidades ou os desejos do usuário a respeito de um produto ou uma funcionalidade (Mannila, 2013). A *story* possui critérios de aceite para definir no momento de seu término, se ela foi implementada corretamente ou não (Pinna *et al.*, 2003). O indicador *leadtime* de *story* é contabilizado do momento em que a *story* entra em refinamento até o momento de sua ativação, ou seja, quando seu escopo foi finalizado. O *leadtime* apresentou melhoria após a reestruturação, diminuindo a quantidade de dias necessários para ativação de uma *story*. O indicador *throughput* de *story* indica a quantidade de *stories* concluídas em um determinado período de tempo. Esse indicador aparentemente se manteve com o mesmo comportamento, porém é interessante notar que após a reestruturação, os dois *squads* somados ativaram praticamente o dobro de *stories* do que se fazia no período anterior.

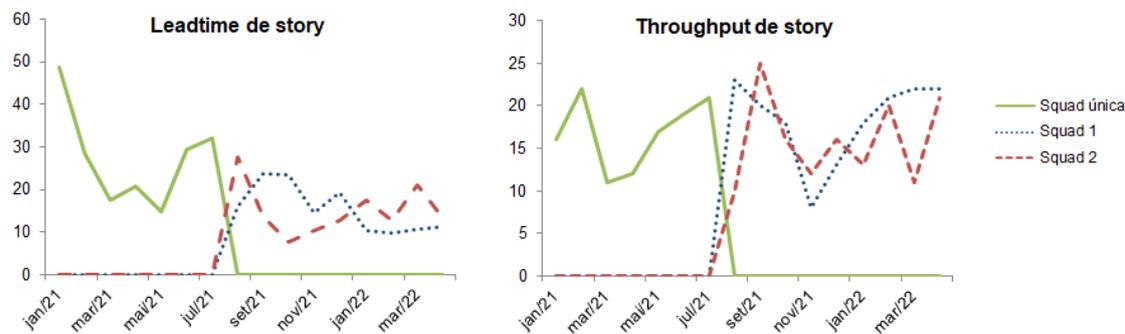


Figura 3 – *Leadtime* de *story* e *throughput* de *story*

Fonte: elaborado pelos autores.

REESTRUTURAÇÃO DE UM *SQUAD* DE DESENVOLVIMENTO DE *SOFTWARE* ÁGIL EM DOIS NOVOS *SQUADS*: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA

As mudanças na área de desenvolvimento de *software* são entendidas como uma melhoria no desempenho de algum *software* existente ou uma nova funcionalidade implantada, caracterizando o número de implantações por mês (Mannila, 2013). O indicador com o nome quantidade de mudanças reflete quantas implantações de *software* foram realizadas em determinado período. O indicador quantidade de falhas nas mudanças mostra quantas dessas implantações não tiveram êxito e acabaram sendo canceladas. Uma falha é um defeito que a equipe tem responsabilidade de corrigir (Mannila, 2013). Para esses dois indicadores, os resultados foram diferentes nos dois *squads* após a reestruturação, conforme demonstra a Figura 4. A quantidade de mudanças implantadas em produção no *squad* 1 foi um pouco maior do que no time unificado. Já no *squad* 2, a quantidade de mudanças foi inferior à equipe antes da quebra. Houve um aumento na quantidade de falhas nas mudanças do *squad* 1 nos quatro meses posteriores a separação das equipes.

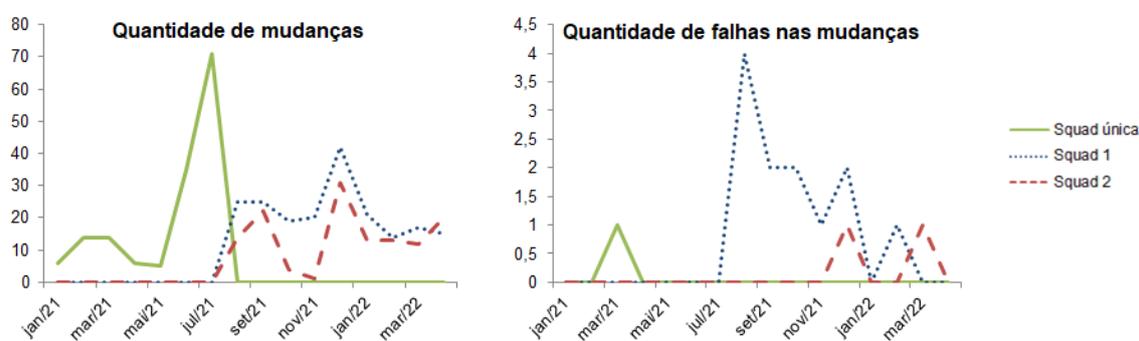


Figura 4 – Quantidade de mudanças e quantidade de falhas nas mudanças
Fonte: elaborado pelos autores.

Como resultado das entrevistas, durante a codificação aberta foram encontrados 22 códigos, com o total de 129 ocorrências. A Tabela 2 mostra como os códigos foram separados em quatro dimensões: (i) benefícios, (ii) competências, (iii) fatores a melhorar e (iv) ferramentas. A primeira dimensão, benefícios, diz respeito aos benefícios obtidos após a quebra do time em dois *squads*. O código com maior número de ocorrências dessa dimensão é o “Jornada do Produto”.

Tabela 2 – Códigos agrupados em dimensões e o número de ocorrências

Dimensões	Código	Nº ocorrências
Benefícios	Jornada do produto	17
	Maturidade	9
	Entregas mais rápidas	5
	Demanda crescente do negócio	4
	Ganho de velocidade para resolver problemas	4
	Tamanho da equipe	4
	Aproximação de negócio	3
	Facilidade em priorização dos projetos	3
	Ganho de confiança	1
Competências	Comunicação	12
	Aumento de engajamento	8
	Especialização em um assunto	8

REESTRUTURAÇÃO DE UM *SQUAD* DE DESENVOLVIMENTO DE *SOFTWARE* ÁGIL EM DOIS NOVOS *SQUADS*: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA

	Ir em busca do conhecimento	7
	Passagem de conhecimento	7
	Colaboração	4
	Time multidisciplinar	1
<hr/>		
	Falta de líder técnico	11
Fatores a melhorar	Transparência na comunicação	9
	Falta de treinamento para novo papel	3
	Projetos que estão no <i>squad</i> errado	2
<hr/>		
Ferramentas e práticas	Cerimônias	4
	Indicadores	3

Fonte: elaborado pelos autores.

A segunda dimensão, competências, abordou algumas competências que já existiam na equipe e que, com a quebra, ficaram mais evidentes. A “Comunicação” foi o código mais mencionado dentro dessa dimensão. Sobre a terceira dimensão, fatores a melhorar, o código mais mencionado foi “Falta de líder técnico”. Por fim, a última dimensão, ferramentas e práticas, teve o código “Cerimônias” com maior aparição.

5 DISCUSSÃO

Para verificar se a reestruturação do *squad* obteve o resultado esperado, algumas métricas do período de antes e depois da reestruturação foram analisadas. Iniciando pela produtividade (figura 2) é possível perceber que os *squads* se tornaram mais produtivos logo no primeiro mês após a sua quebra. Apesar da melhoria em destaque, é importante ressaltar que os dois *squads* seguiram metodologias diferentes de contagem no período em análise. O *squad* 1 contou os BCPs para quase todas as atividades desempenhadas pela equipe. Já o *squad* 2 seguiu uma metodologia mais restrita para contagem, como era feito na equipe antes da reestruturação.

O *throughput* de BCP (figura 2) mede a quantidade de *business complexity points* ativados no período. Para esse indicador, quanto mais pontos ativados, melhor o resultado da equipe. É possível perceber que esse indicador teve uma melhoria a partir da quebra dos *squads*. Tanto a produtividade, quanto o *throughput* de BCP apresentaram evolução por alguns motivos: os times separados passaram a ter uma maior produtividade, o tema de contagem de BCPs ficou mais evidente nas equipes e a empresa lançou um portal de contagem de BCPs que ajudou os desenvolvedores na contagem desses pontos.

Apesar da melhoria em destaque, é importante ressaltar que os dois *squads* seguiram metodologias diferentes de contagem no período em análise. O *squad* 1 contou os BCPs para quase todas as atividades desempenhadas pela equipe. Já o *squad* 2 seguiu uma metodologia mais restrita para contagem, como era feito na equipe antes da reestruturação.

No período de estudo, incentivou-se que as equipes dividissem os escopos dos projetos em entregas menores, de acordo com o conceito de MVP (*Minimum Viable Product*). Essa ação refletiu para que os times criassem as *stories* com escopos menores, diminuindo assim seu tempo de execução e tendo como consequência um menor *leadtime* de *story* conforme apresentado na Figura 3. O indicador *throughput* de *story* (figura 3) indica a quantidade de *stories* concluídas em um determinado período de tempo. Em relação ao *throughput* de *story*, o número se manteve praticamente igual ao que era antes e após a quebra. Em números absolutos, o número de *stories* cresceu ao somar o resultado dos *squads* e comparar com o período anterior à separação.

REESTRUTURAÇÃO DE UM *SQUAD* DE DESENVOLVIMENTO DE *SOFTWARE* ÁGIL EM DOIS NOVOS *SQUADS*: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA

Em relação à quantidade de falhas nas mudanças (figura 4), devido a um movimento da instituição financeira no período em estudo, houve uma migração da baixa plataforma para nuvem. Esse movimento foi intensificado a partir do segundo semestre/2021, por isso a quantidade de falhas não necessariamente reflete um resultado da reestruturação dos times, mas sim de uma adaptação à tecnologia utilizada por eles.

Como resultado das entrevistas foi elaborada a tabela 2. Os códigos foram separados em quatro dimensões: (i) benefícios, (ii) competências, (iii) fatores a melhorar e (iv) ferramentas. A primeira dimensão, benefícios, teve o código “Jornada do Produto” com o maior número de ocorrência. Sobre isso, os times foram quebrados em duas jornadas do cheque especial: a contratação e a manutenção. Com essa estratégia, foi possível que os *squads* se especializassem cada uma em seu assunto. O entrevistado E3 afirma: “tendo um escopo definido do que seu *squad* vai atuar, sabemos o que precisa entregar. Consequentemente somos mais objetivos, limitados o escopo e deixamos de fazer o que não é necessário”. O entrevistado E5 complementa: “eu vejo que essa quebra é benéfica, conseguindo focar em projetos da nossa alçada, fica mais específico”. Por fim, o entrevistado E12 complementa: “com menos assuntos para serem tratados, o time consegue focar em questões mais específicas e ter domínio de algo que esteja em seu *squad*”. O achado vai ao encontro dos resultados de Dingsøyr e Lindsjørn (2013), aonde a priorização das tarefas da equipe foi percebida como um dos fatores mais importantes para alcançar seu melhor desempenho.

A segunda dimensão, competências, obteve o código “Comunicação” com o maior número de ocorrência. O entrevistado E1 abordou: “com o time grande ficava difícil de se comunicar [...], a fluidez era difícil no dia a dia. Então acho que reduzindo o time melhorou. [...] Ficou muito mais ágil a comunicação.”. Para Rotta *et al.* (2015) a comunicação se torna mais eficiente e constante em equipes de alta performance, o que promove melhoria do clima organizacional e aumento da produtividade. Neste sentido, o entrevistado E5 complementou: “em um time um pouco menor, um pouco mais restrito, você tende a ter menos interferências na comunicação.”.

Por fim, o entrevistado E8 mencionou: “é muito mais fácil você administrar o dia a dia de um *squad* com oito pessoas do que com dezesseis, a comunicação fica muito mais fácil, a chance de um retrabalho é menor.”. Segundo Gonçalves (2020), o *squad* é projetado para dar maior velocidade no desenvolvimento dos produtos e na prestação de serviços da empresa, caracterizando-se por grupos de 3 a 10 membros com características complementares capazes de desenvolver o projeto em sua completude.

Sobre a terceira dimensão, fatores a melhorar, o código com maior ocorrência foi “Falta de líder técnico”. Quando os times foram separados, um deles ficou sem esse papel formalmente, embora o líder técnico do time pré-divisão tenha colaborado com os dois times durante um período de três meses. Com relação a este aspecto, o entrevistado E4 explicou: “Quando houve a quebra, um time ficou mais equipado e o outro mais defasado. Os membros que estavam lá foram obrigados a assumir a responsabilidade de líder técnico, eu senti um desenvolvimento muito forte deles. Eles passaram a assumir papéis que talvez antes eles não assumissem porque eles estavam no meio de muita gente. Então nesse sentido teve efeito positivo.”. Corroborando com esta ideia, Hoegl e Gemuenden (2001) afirmam que o desempenho da equipe de desenvolvimento de *software* é influenciado pelo suporte mútuo e comunicação.

O entrevistado E5 mencionou: “acho que seria interessante se tivessem preparado o *capacity* [alocação de recursos nas vagas disponíveis] antes de fazer a quebra. Quando quebrou, um dos *squads* não tinha líder técnico, faltou *team member* [membros da equipe]. Então eu acho que a gestão podia ter preparado esse *capacity* antes de fazer a quebra.” Por fim, o entrevistado E9 conclui: “a grande dificuldade foi o líder técnico, não tínhamos duas pessoas prontas para

REESTRUTURAÇÃO DE UM *SQUAD* DE DESENVOLVIMENTO DE *SOFTWARE* ÁGIL EM DOIS NOVOS *SQUADS*: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA

esse papel, então uma dos *squads* nasceu sem esse papel e foi um risco que a gente combinou de comprar juntos com prazo a ser resolvido.”

Por fim, a última dimensão, ferramentas e práticas, teve o código “Cerimônias” com maior número de ocorrência. O maior benefício aqui encontrado foi a maior participação do time nas cerimônias, assim como a redução do tempo necessário para sua execução. O entrevistado E4 mencionou: “eu senti que as cerimônias ficaram mais rápidas e isso me marcou bastante. A *daily* antes acontecia de 30 a 40 minutos, aí ela passou a ter 10 minutos”. O entrevistado E7 conclui: “antes tínhamos um mix de *backlog*, vários objetivos no *squad* [...], fazendo com que as pessoas não prestassem tanta atenção na *daily*, na *review*.”

No sentido das reduções propostas e discutidas até aqui, Gonçalves *et al.* (2020) mencionam que equipes mais enxutas provaram ser mais produtivas, colaborativas, com baixo custo de comunicação e melhor gerenciadas. Portanto, a reestruturação proposta em duas *squads* proporcionou benefícios às equipes, que também são convertidos em resultados positivos à empresa.

6 CONCLUSÕES E CONSIDERAÇÕES FINAIS

Este estudo buscou responder a seguinte pergunta de pesquisa: “Quais as consequências na performance decorrentes da decomposição de uma equipe de desenvolvimento de software usando abordagem ágil em duas equipes em uma instituição financeira brasileira?”. Entende-se a relevância do tema, visto a crescente demanda mundial de desenvolvimento de *software* e aumento do uso de abordagem ágil.

O objetivo esperado foi atingido e o estudo apresentou as consequências da decomposição de uma equipe de desenvolvimento de *software* em dois *squads*. Os novos *squads* tiveram uma melhor performance por interferência de vários fatores: ganho de benefícios, aumento de competências e melhor uso das ferramentas e práticas.

Este trabalho também levantou, por meio das entrevistas, alguns fatores que podem ser melhor explorados se houver necessidade de decompor um time, por exemplo: preparar e disponibilizar todos os papéis que o compõe; comunicar os membros com transparência a respeito do processo da quebra, incluindo os fatores motivadores e o método utilizado para divisão dos times; treinar os novos integrantes que assumirão papéis diferentes do que desempenhavam antes; e alocar os projetos no *squad* correto de acordo com o escopo da mesma. Na prática, esse trabalho contribui com algumas possíveis diretrizes que podem ser adotadas pelas empresas no momento de reestruturar seus *squads* e conseguir um melhor desempenho nessa tarefa.

As limitações deste estudo consistem na aplicação do estudo apenas em uma instituição financeira privada e no número reduzido de participantes. Assim, para estudos futuros sugere-se a análise da reestruturação de equipes em diferentes setores e empresas para verificação dos impactos. Como proposta futura para pesquisa, os autores pretendem entender como se dá a formação de times de alta performance em um contexto de gerenciamento de projetos ágeis. Nessa mesma linha, outros pesquisadores podem observar as diferenças em relação ao tipo de projeto e abordagens utilizados.

7 REFERÊNCIAS

Ahmad, M., Abdulmajeed, V., Omar, M., Yasin, A., Baharom, F., Mohd, F., & Darus, N. M. (2016). Examining the Influence of Team Work Factors on Team Performance for

**REESTRUTURAÇÃO DE UM SQUAD DE DESENVOLVIMENTO DE SOFTWARE ÁGIL EM DOIS
NOVOS SQUADS: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA**

- Software Development in Telecommunication Industry. Journal of Telecommunication, Electronic and Computer Engineering*, 8 (8), 121 – 124.
- Almeida, C. C., Campagnac, L. A. & Farias Filho, J. R. (2008). Em busca do perfil ideal de gerente para alcançar o sucesso dos projetos. *Congresso Nacional de Excelência em Gestão*, 4.
- Amaral, D. C., Conforto, E. C., Benassi, J. L. G., & Araújo, C. (2011). Gerenciamento ágil de projetos: aplicação em produtos inovadores. São Paulo: Saraiva.
- Barki, H. & Hartwick, J. (2004). Conceptualizing the construct of interpersonal conflict. *International Journal of Conflict Management*, 15, (3), 216-44.
- Biancolino, C. A., Kniess, C. T., Maccari, E. A., & Rabechini Jr., R. (2012). Protocolo para Elaboração de Relatos de Produção Técnica. *Revista de Gestão e Projetos*, 3(2), 294–307.
- Brodbeck, F.C. (2001). Communication and performance in *software* development projects. *European Journal of Work and Organizational Psychology*, 10, (1), 73–94.
- Bryman, A. (1989). *Research methods and organization studies*. Londres: Unwin Hyman.
- Charmaz, K. (2006). *Constructing Grounded Theory: A Practical guide through qualitative analysis*. London. Sage.
- Cizmaş, E., Feder, E. S., Maticiu, M. D., & Vlad-Anghel, S. (2020). Team management, diversity, and performance as key influencing factors of organizational sustainable performance. *Sustainability*, 12(18), 7414.
- Crispin, L., & Gregory, J. (2009). *Agile testing: A practical guide for testers and agile teams*. Boston, MA: Pearson Education.
- Data Driven Agile. (2015). <http://datadrivenagile.blogspot.com/>
- De Barros Sampaio, S. C., Barros, E. A., de Aquino, G. S., e Silva, M. J. C., & de Lemos Meira, S. R. (2010, August). A review of productivity factors and strategies on software development. In *2010 fifth international conference on software engineering advances* (pp. 196-204). IEEE.
- De Medeiros, A. A. (2011). O processo de definição do escopo do projeto segundo o PMBOK. *Revista de Ciências Gerenciais*, 15(21).
- De Moraes, A. T., da Silva, L. F., & de Oliveira, P. S. G. (2020). Systematization of absorptive capacity microprocesses for knowledge identification in project management. *Journal of Knowledge Management*, 24 (9), 2195 – 2216.
- del-Río-Ortega, A., Resinas, M., Cabanillas, C., & Ruiz-Cortés, A. (2013). On the definition and design-time analysis of process performance indicators. *Information Systems*, 38(4), 470-490.
- Dingsøy, T. & Lindsjørn, Y. (2013). Team performance in agile development teams: findings from 18 focus groups. *Agile Processes in Software Engineering and Extreme Programming*, 46–60.
- Doherty, N. F., Ashurst, C., & Peppard, J. (2012). Factors affecting the successful realisation of benefits from systems development projects: findings from three case studies. *Journal of Information technology*, 27(1), 1-16.
- Dybå, T., & Dingsøy, T. (2008). Empirical studies of agile software development: A systematic review. *Information and software technology*, 50(9-10), 833-859.
- Fagerholm, F., Ikonen, M., Kettunen, P., Münch, J., Roto, V., & Abrahamsson, P. (2015). Performance Alignment Work: How *software* developers experience the continuous adaptation of team performance in Lean and Agile environments. *Information and Software Technology*, 64, 132-147.

**REESTRUTURAÇÃO DE UM SQUAD DE DESENVOLVIMENTO DE SOFTWARE ÁGIL EM DOIS
NOVOS SQUADS: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA**

- Fleury, M. T. L., & Fleury, A. (2001). Construindo o conceito de competência. *Revista de Administração Contemporânea*, 5(spe), 183–196.
- Fung, H. P. & Ali, I. (2011). Factors influencing project team effectiveness as perceived by project managers in malaysia: A pilot study. *Annual Summit on Business and Entrepreneurial Studies*, 1–20.
- FRIESE, S. (2012). *Qualitative data analysis with Atlas.ti*. London. Sage.
- Ganis, M. R., & Waszkiewicz, M. (2018). Digital Communication Tools as a Success Factor of Interdisciplinary Projects. *Problemy Zarzadzania*, 16.
- Gonçalves, L. C. C., Oliveira, S. A. A., Pacheco, J. C. A., & Salume, P. K. (2020). Competências requeridas em equipes de projetos ágeis: um estudo de caso em uma Edtech. *Revista de Gestão e Projetos (GeP)*, 11(3), 72-93.
- Grander, G., Dal Vesco, D. G., & Ribeiro, I. (2019). The effect of project governance and performance management on organizational strategy: a multi-group analysis under the prism of performance indicators. *Revista de Gestão e Projetos*, 10(2).
- Hoda, R., Salleh, N., & Grundy, J. (2018). The rise and evolution of agile software development. *IEEE software*, 35(5), 58-63.
- Hoegl, M. & Gemuenden, H. G. (2001). Teamwork quality and the success of innovative projects: A theoretical concept and empirical evidence. *Organization Science*, 12(4), 435–449.
- Hsu, J. S.-C., Shih, S.-P., Chiang, J. C., & Liu, J. Y. C. (2012). The impact of transactive memory systems on IS development teams' coordination, communication, and performance. *International Journal of Project Management*, 30(3), 329–340.
- Jiang, J. J., Klein, G., Wu, S. P., & Liang, T. P. (2009). The relation of requirements uncertainty and stakeholder perception gaps to project management performance. *Journal of Systems and Software*, 82(5), 801-808.
- Katzenbach, J. & Smith, D. (1993). The discipline of teams. *Harvard Business Review*, 71(2), 111-120.
- Lechler, T. (1997). Success factors of project management. *Frankfurt am Main*.
- Levi, D. & Slem, C. (1995). Team work in research and development organizations: The characteristics of successful teams. *International Journal of Industrial Ergonomics*, 16, 29-42.
- Liang, T.; Liu, C.; Lin, T. & Lin, B. (2007). Effect of team diversity on software project performance. *Industrial Management & Data Systems*, 107 (5), 636-653.
- Lindsjörn, Y., Sjøberg, D.I.K., Dingsøy, T., Bergersen, G.R. & Dybå, T. (2016). Teamwork quality and project success in software development: A survey of agile development teams. *J. Syst. Softw.*, 122, 274–286.
- Lu, M.; Watson-Manheim, M. B.; Chudoba, K. M. & Wynn, E. (2006). Virtuality and Team Performance: Understanding the Impact of Variety of Practices. *Journal of Global Information Technology Management*, 9, (1), 4-23.
- Machuca-Villegas, L., Gasca-Hurtado, G. P., & Muñoz, M. (2021). Measures related to social and human factors that influence productivity in software development teams. *International Journal of Information Systems and Project Management*, 9(3), 43-67.
- Mannila, J. (2013). Key performance indicators in agile software development.
- McClough, A. C. & Rogelberg, S. G. (2003). Selection in teams: An exploration of the teamwork knowledge, skills and ability test. *International Journal of Selection and Assessment*, 11(1), 56-66.
- Melo, C., Cruzes, D.S., Kon, F., & Conradi, R. (2011). Agile team perceptions of productivity factors. *Proc. 2011 Agile Conf. IEEE Computer Society*, 57–66.

REESTRUTURAÇÃO DE UM SQUAD DE DESENVOLVIMENTO DE SOFTWARE ÁGIL EM DOIS NOVOS SQUADS: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA

- Moe, N. B., Cruzes, D. S., Dybå, T., & Engebretsen, E. (2015, July). Coaching a global agile virtual team. In *2015 IEEE 10th International Conference on Global Software Engineering* (pp. 33–37). Ciudad Real, Spain: IEEE.
- Paletta, F.C. & Dias Vieira, N., Jr. (2008). Information technology and communication and best practices in it lifecycle management. *J. Technol. Manag. Innov.*, 3, 80–94.
- Palopak, Y., & Huang, S. J. (2022). Correlation of Agile Principles and Practices to Software Project Performance: An AHP–Delphi Analysis. *International Journal of Software Engineering and Knowledge Engineering*, 32(02), 257-281.
- Patanakul, P. & Shenhar, A. J. (2012). What Project Strategy Really Is: The Fundamental Building Block in Strategic Project Management. *Project Management Journal*, 43(1), 4–20.
- Piccoli, G., Powell, A., & Ives, B. (2004). Virtual teams: team control structure, work processes, and team effectiveness. *Information Technology & People*.
- Pikkarainen, M.; Haikara, J.; Salo, O.; Abrahamsson, P.; & Still, J. (2008). The impact of agile practices on communication in software development. *Empir. Softw. Eng.*, 13, 303–337.
- Pinna, S., Lorrai, P., Marchesi, M., & Serra, N. (2003, August). Developing a tool supporting XP process. In *Conference on Extreme Programming and Agile Methods* (pp. 151-160). Springer, Berlin, Heidelberg.
- Project Management Institute (2008). *A guide to the Project Management Body of Knowledge: PMBOK® Guide* (4a ed.). Pennsylvania: PMI.
- Project Management Institute. (2013). *Software Extension to the PMBoK Guide*. Project Management Institute.
- Project Management Institute (2021). *A guide to the Project Management Body of Knowledge: PMBOK® Guide* (7a ed.). Pennsylvania: PMI.
- Rabechini Junior, R., & Carvalho, M. M. D. (2003). Perfil das competências em equipes de projetos. *RAE eletrônica*, 2.
- Rabechini Jr., R.; Carvalho, M. M.; Rodrigues, I. & Sbragia, R. (2011). A organização da atividade de gerenciamento de projetos: os nexos com competências e estrutura. *Gest. Prod.*, 18 (2), 409-424.
- Richards, B. (2003). Intelligent teams: The dynamics of collaboration. *Association for Quality & Participation*, 20-22.
- Rotta, M., Rados, G. J. V., Trierweiler, A. C., & Silva, S. M. (2015). Implantação de métodos ágeis em empresa de desenvolvimento de software: um estudo de caso. *Congresso do Conhecimento e Inovação*, 5.
- Russo, R. D. F. S. M., & da Silva, L. F. (2019). Aplicação de entrevistas em pesquisa qualitativa. *Gestão e Projetos: GeP*, 10(1), 1-6.
- Sefiani, N., Boumane, A., Campagne, J. P. & Bouami, D. (2012). Process of identifying competencies based on a functional approach. *International Journal of Engineering Science and Technology*, 4(1), 265-275.
- Serra, C. E. M., & Kunc, M. (2015). Benefits Realisation Management and its influence on project success and on execution of business strategies. *International Journal of Project Management*, 33, 53–66.
- Sharp, H. & Robinson, H. (2010). Three “C”s of agile practice: Collaboration, co-ordination and communication. *Agile Software Development*, 61–85.
- Stray, V., Sjøberg, D.I.K., Dybå, T., 2016. The daily stand-up meeting: a grounded theory study. *J. Syst. Softw.* 114, 101–124.
- Sudhakar, G. P. (2010). Teamwork in software organizations. *CSI Communications*, 34, 22–24.
- Sudhakar, G.P. (2012). A model of critical success factors for software projects. *Journal of Enterprise Information Management*, 25(6), 537–558.

REESTRUTURAÇÃO DE UM SQUAD DE DESENVOLVIMENTO DE SOFTWARE ÁGIL EM DOIS NOVOS SQUADS: UM ESTUDO DE CASO EM UMA INSTITUIÇÃO FINANCEIRA

- Sutherland, J. & Schwaber, K. (2013). Um guia definitivo para o Scrum: as regras do jogo. Guia do Scrum. Recuperado em 11 novembro, 2018, de <https://www.scrumguides.org/docs/scrumgui de/v1/Scrum-Guide-Portuguese-BR.pdf>.
- THAMHAIN, H. J. Team Building in Project Management. In: CLELAND, D. I.; KING, W. R. Project Management Handbook. New York: Van Nostrand Reinhold, 1993.
- Tirmizi, S.A. (2008). Towards Understanding Multicultural Teams. In Effective Multicultural Teams: Theory and Practice. *Advances in Group Decision and Negotiation*, 3, 1–20.
- Zheng, M., Zada, I., Shahzad, S., Iqbal, J., Shafiq, M., Zeeshan, M., & Ali, A. (2021). Key performance indicators for the integration of the service-oriented architecture and scrum process model for IOT. *Scientific Programming*, 2021.
- Yang, C., Liang, P., & Avgeriou, P. (2016). A systematic mapping study on the combination of software architecture and agile development. *Journal of Systems and Software*, 111, 157-184.
- Yusoff, M. Z., Mahmuddin, M., & Ahmad. (2012). A conceptual model of knowledge work productivity for software Development Process: Quality issues. *Knowledge Management International Conference (KMICe)*, 334–342.

Apêndice A

O roteiro das entrevistas semiestruturadas foi composto pelas seguintes questões:

1. Quais foram os fatores motivadores da reestruturação das *squads*?
2. O que era esperado como resultado da reestruturação?
3. Em relação à aplicação das competências do time, elas mudaram se comparar a equipe antes da reestruturação e após sua quebra em dois novos *squads*?
4. Os resultados individuais e do time foram alterados com a intervenção realizada?
5. A satisfação dos integrantes da equipe foi impactada com a mudança?
6. Pessoalmente como foi essa reestruturação para sua rotina?
7. Como você lidou com a mudança?
8. No entendimento da empresa, os novos *squads* estão no patamar de alta performance?
9. A forma que a equipe lida com os problemas foi alterada após a reestruturação?
10. O desenvolvimento de competências da equipe foi beneficiado com a intervenção?
11. O engajamento da equipe foi beneficiado com a intervenção?
12. Se a reestruturação acontecesse novamente, quais pontos poderiam ser feitos diferentemente para alcançar um melhor resultado?