



A RELAÇÃO ENTRE A UTILIZAÇÃO DA METODOLOGIA SCRUM E A GARANTIA DE QUALIDADE DE SOFTWARE DEFINIDA PELO MODELO ISO 25010: UMA ANÁLISE COMPARATIVA NA PERSPECTIVA DE UMA ORGANIZAÇÃO

Submetido em: 22/11/2020

Aprovado em: 05/12/2022

ISSN 2965-3339

DOI: 10.29327/2384439.1.2-1

Nilson José Moreira Filho

Faculdade de Tecnologia Zona Leste
nilsonjosemoreirafilho@gmail.com

Yago Henrique dos Santos Rodrigues

Faculdade de Tecnologia Zona Leste
yagohts@hotmail.com

Luciano Francisco de Oliveira

Faculdade de Tecnologia Zona Leste
luciano.oliveira@fatec.sp.gov.br

RESUMO.

O presente artigo tem como intuito verificar possíveis relações entre a utilização da metodologia Scrum e o conceito de qualidade de software, baseado no modelo de qualidade de produto de software fornecido pela ISO 25010, em uma perspectiva organizacional. O objetivo geral é identificar, por meio de análise comparativa entre equipes de desenvolvimento de software, relações entre a adoção da metodologia de desenvolvimento de software e a aderência às diretrizes de qualidade estabelecidas pelo modelo citado. Para tanto, conceituou-se qualidade de software, o modelo de qualidade de produto de software da ISO 25010, a metodologia Scrum, apresenta-se as características da organização e as equipes adotadas como objeto de estudo, se descreve a aplicação da coleta de dados primários e realiza-se discussão sobre os resultados obtidos. O tema se justifica pelo aumento da exigência por qualidade de software, além da popularização da metodologia Scrum, o que torna interessante a busca por relações entre os dois conceitos em um ambiente de desenvolvimento de software real. A partir de um estudo de caráter descritivo, consistindo em uma análise comparativa embasada em fontes primárias e secundárias e abordagem qualitativa e quantitativa dos dados, demonstra-se que, no caso estudado, apesar das hipóteses apontarem que o Scrum não seria influente na maioria dos pilares do modelo, a chamada de perspectiva “Não Scrum”, mostrou melhores graus de qualidade. Além disso, raras foram as questões em que não foi demonstrado certo distanciamento entre as perspectivas. Outro resultado relevante é que características presentes no Scrum se mostraram malélicas em alguns quesitos do modelo, como a manutenibilidade.

Palavras-chave. Scrum, Qualidade, ISO 25010.

ABSTRACT.

This article aims to verify possible relationships between the use of the Scrum methodology and the concept of software quality, based on the software product quality model provided by ISO 25010, in an organizational perspective. The general objective is to identify, through comparative analysis between software development teams, relationships between the adoption of the methodology and adherence to the quality guidelines established by the model mentioned. For that, software quality, the ISO 25010 software product quality model and the Scrum methodology are conceptualized, the organization and teams adopted as the object of study are presented, the application of primary data collection is described and there is a discussion about the results obtained. The theme is justified by the increased demand for software quality, in addition to the popularization of the Scrum methodology, which makes it interesting to search for relationships between the two concepts in a real software development environment. From a descriptive study, consisting of a comparative analysis based on primary and secondary sources and a qualitative and quantitative approach to the data, it is demonstrated that, in the case studied, despite the hypotheses pointing out that Scrum would not be influential in most cases. pillars of the model, the so-called “Non-Scrum” perspective, showed better degrees of quality. In addition, there were very few issues where a certain distance between the perspectives was not demonstrated. Another relevant result is that features present in Scrum proved to be harmful in some aspects of the model, such as maintainability.

Keywords. Scrum, Quality, ISO 25010.

1. INTRODUÇÃO

A constante busca pela evolução dos processos no mercado de desenvolvimento de software proporcionou o surgimento de diversas técnicas e metodologias. Dentre elas se destacam as metodologias ágeis, com propostas de redução de prazo e custos do projeto, entre outras, como é o caso da metodologia Scrum, que tem se popularizado nos âmbitos acadêmico e corporativo.

Em paralelo, a preocupação com a qualidade de software produzido vem crescendo, de forma que garantir a qualidade do produto se mostrou tão importante quanto sua codificação (COLLINS et al, 2010). Neste sentido se mostra interessante a adoção de modelos de qualidade de software, por exemplo, o modelo de qualidade de produto de software (MQPS) da ISO 25010, como ferramenta auxiliadora em projetos de software.

Discutir e verificar possíveis relações entre a adoção do Scrum como metodologia de desenvolvimento e o MQPS da ISO 25010, principalmente em um ambiente real do mercado de desenvolvimento, justifica-se pelo fato de que cada vez mais, dada a popularidade do framework, tal como suas propostas, as organizações vêm utilizando o Scrum para tentar aprimorar seu processo de desenvolvimento, contudo, em concorrência, precisam se preocupar com a qualidade de seus produtos, uma vez que os produtos de software são utilizados cada vez mais, em domínios cada vez mais críticos e complexos. Com isso, se define o seguinte problema de pesquisa: Existem relações entre a adoção da metodologia Scrum, em um contexto organizacional, e a qualidade do software produzido baseada nos pilares descritos pelo MQPS da ISO 25010?

Definida a questão central da pesquisa, estabelece-se como objetivo geral: Identificar, por meio de análise comparativa entre equipes de desenvolvimento de

software, possíveis relações entre a adoção da metodologia e a aderência às diretrizes de qualidade de software estabelecidas pelo MQPS definido pela ISO 25010 e, além disso, os seguintes objetivos específicos:

- Conceituar qualidade de software e o MQPS apresentado pela ISO 25010.
- Conceituar a metodologia Scrum.
- Descrever o cenário das equipes de desenvolvimento de software que são utilizadas como objeto de estudo.
- Descrever os questionários e entrevistas aplicadas.
- Discutir os resultados obtidos buscando relações entre a utilização da metodologia Scrum e o MQPS definido pela ISO 25010.

Visando alcançar os objetivos definidos, o presente estudo consiste em pesquisa aplicada de caráter descritivo. Os resultados são apresentados de forma qualitativa e quantitativa a partir de coleta de dados e informações provenientes de fontes primárias e secundárias.

Como fontes secundárias, são considerados trabalhos de diversos autores com o intuito de promover uma revisão consistente sobre os conceitos chave do tema. As fontes primárias, utilizadas na busca pela solução do problema de pesquisa apresentado, sendo baseadas em questionários e entrevistas.

Os questionários são aplicados a integrantes de equipes de desenvolvimento em uma empresa que produz software para o setor financeiro, com perguntas relacionadas aos pilares do MPQS da ISO 25010. As entrevistas são realizadas com líderes das equipes anteriormente citadas, e possuem perguntas abertas, ainda em relação ao MPQS da ISO 25010, com o intuito de obter informações de abordagem qualitativa.

Desta forma, o trabalho consiste em uma análise crítica, que utiliza os dados obtidos por meio das fontes citadas, visando

identificar possíveis relações entre a utilização do Scrum, em uma organização, e os pilares do MPQS definido pela ISO 25010.

O presente estudo encontra-se segmentado em cinco capítulos, sendo o primeiro deles esta introdução. Em seguida, o segundo capítulo apresenta a fundamentação teórica em relação aos conceitos chave do trabalho, Scrum e o MPQS da ISO 25010. O terceiro capítulo apresenta os materiais e métodos utilizados no desenvolvimento do presente artigo. O quarto capítulo apresenta os resultados obtidos por meio das fontes primárias, sua análise e discussão. Por fim, o quinto capítulo apresenta as considerações finais do estudo.

2. FUNDAMENTAÇÃO TEÓRICA

O presente capítulo aborda de forma teórica os conceitos chave do artigo: Qualidade de Software e o MPQS da ISO 25010; e a metodologia Scrum.

2.1. QUALIDADE DE SOFTWARE

A qualidade, quando atrelada a sistemas de informação, tem suas peculiaridades e definições que merecem atenção maior. A partir disso, em meio à Engenharia de Software, apresenta-se a qualidade de software como o estudo das variantes qualitativas específicas à produção de softwares. Segundo Pressman (2010), a qualidade de software pode ser definida como “uma gestão de qualidade efetiva aplicada de modo a criar um produto útil que forneça valor mensurável para aqueles que o produzem e para aqueles que o utilizam”. Outra definição é da qualidade como algo dependente do grau de satisfação das necessidades dos clientes sob todos os aspectos do produto (SANDERS; CURRAN, 1994). Sambo (2016) destaca a necessidade da eficácia do software, a utilidade do produto e geração de valor. Para todas as definições, alguns pontos em

comum são importantes. Destacam-se os alinhamentos a requisitos de software definidos, a processos de desenvolvimento e a requisitos implícitos.

É importante salientar que para um software obter qualidade, seus requisitos e características devem ser bem definidos. Dificilmente a qualidade pode ser incorporada ao produto, normalmente advém de práticas tomadas no processo de desenvolvimento. A partir disso, mesmo sendo complementares e dependentes, pode-se considerar os seguintes dois conceitos dentro da qualidade de software: a qualidade de processo e a qualidade de produto de software (TSUKUMO et al, 1997).

A qualidade de processo de software pode ser atribuída a fatores de atendimento aos requisitos em âmbito de definição das atividades a serem realizadas no desenvolvimento do software.

A qualidade de produto de software está relacionada a fatores característicos de avaliação do produto de software, fatores esses que são bem abrangentes, como características relacionadas tanto ao atendimento aos requisitos quanto à satisfação do cliente. Segundo Bartié (2002 pp. 16-18) propõe a importância de associar a dimensão da qualidade do processo e da qualidade do produto.

Em complemento à qualidade de processo de software, a qualidade de produto de software é fator determinante para o alcance da qualidade desejada para um software.

Dadas as várias características que desencadeiam em um produto de software de qualidade, a discussão de conceitos e modelos a serem utilizados para o desenvolvimento de um projeto de software é necessária. Para mensurar a qualidade de um produto de software, deve-se traçar uma diretriz avaliativa sobre o projeto. Objetivando que se consiga avaliar um produto de software de forma efetiva, tem-se no mercado alguns modelos de qualidade

de produto, como o modelo de McCall, o modelo da Hewlett Packard e as definições das normas ISO/IEC 9126 que foi reestruturada na série ISO/IEC 25010 (GUERRA; COLOMBO, 2009).

2.1.1 ISO/IEC 25010

Para Mauda (2012) A família dessa norma é chamada ISO/IEC 25000 proveniente do projeto SQuaRE (*Software Product Quality Requirements and Evaluation*), que compõem uma família de normas estruturadas, visando cobrir dois processos principais: a especificação de requisitos e a avaliação da qualidade de software, ambos utilizando-se de um processo de medição. Por se tratar de uma família de normas, o projeto SQuaRE foi definido dentro de um intervalo numérico, entre os números 25000 e 25099. Um processo de avaliação de qualidade deve contemplar as várias etapas existentes dentro do ciclo de vida de um software, a norma realizou a divisão deste intervalo.

Segundo a ISO/IEC 25010, a qualidade de um sistema é o grau em que o sistema satisfaz as necessidades declaradas e implícitas das várias partes interessadas e, assim sendo, fornece valor. Essas necessidades declaradas e implícitas são representadas na série *Systems and software Quality Requirements and Evaluation (SQuaRE)* de Padrões Internacionais por modelos de qualidade que categorizam a qualidade do produto em características (ou pilares), que em alguns os casos são subdivididos em subcaracterísticas. Assim, a definição dos modelos apresentados pela norma possui diretrizes para que seja mantido o padrão entre as estruturas.

Em mais especificidade a ISO/IEC 25010 define dois modelos. Um modelo de qualidade no uso composto por cinco pilares, dos quais alguns possuem subcaracterísticas, que são relacionadas ao resultado da interação com o produto em um determinado contexto de uso. O outro, um

modelo de qualidade do produto que é composto por oito pilares, que são subdivididos em subcaracterísticas, relacionados com as propriedades estáticas de um produto de software e dinâmicas de sistemas de computador.

O modelo de qualidade apresentado pela ISO/IEC 25010 no uso apresenta os seguintes pilares: Efetividade, Eficiência, Satisfação, Uso Sem Risco e Cobertura de Contexto. As características atreladas ao modelo de qualidade no uso buscam atender aos principais fatores que contribuem para a avaliação da qualidade na perspectiva do uso.

Já o modelo de qualidade do produto demonstrado pela ISO/IEC 25010 tem como alvo o sistema de computador, no qual está incluso o produto de software, hardware e dados, e apresenta os seguintes pilares para atender as necessidades: Adequação Funcional, Eficiência de Desempenho, Compatibilidade, Usabilidade, Confiabilidade, Segurança, Manutenibilidade e Portabilidade (ISO 25000, acessado em novembro de 2020).

A Adequação Funcional é o grau em que um produto ou sistema fornece funções que atendem às necessidades declaradas e implícitas quando usado sob condições especificadas. Esta característica apresenta três subcaracterísticas que são: Completude Funcional, Assertividade Funcional e Apropriação Funcional.

A Eficiência de Desempenho é o desempenho em relação à quantidade de recursos usados (que podem incluir outros produtos de software, a configuração de hardware e software, entre outros) nas condições estabelecidas. Apresenta três subcaracterísticas, que são: Comportamento no tempo, Utilização de Recursos e Capacidade.

A Compatibilidade é o grau em que um produto, sistema ou componente pode trocar informações com outros produtos, sistemas ou componentes, executar as

funções necessárias, enquanto compartilha o mesmo ambiente de hardware ou software. A compatibilidade apresenta duas subcaracterísticas: Coexistência e Interoperabilidade.

A Usabilidade é o grau em que um produto ou sistema pode ser usado por seus usuários para atingir objetivos específicos com eficácia, eficiência e satisfação em um contexto de uso específico. Possui seis subcaracterísticas, são elas: Adequação em Familiaridade, Aprendizagem, Operabilidade, Proteção a Erro de Usuário, Estética da Interface de Usuário e Acessibilidade.

A Confiabilidade é o grau ao qual um sistema, produto ou componente executa determinadas funções em condições específicas em um período definido. A Confiabilidade possui quatro subcaracterísticas: Maturidade, Disponibilidade, Tolerância a Falhas e Recuperabilidade.

A Segurança é o grau que um produto ou sistema protege as informações e dados, para que pessoas e outros produto tenham acesso a eles em nível adequado aos seus níveis de autorização. A segurança possui cinco subcaracterísticas, são elas: Confidencialidade, Integridade, Não-repúdio, Responsabilidade e Autenticidade.

A Manutenibilidade é o grau de eficácia e eficiência que um sistema ou produto pode ser modificado pelos que o mantém, modificações podem incluir correções, melhorias ou adaptações do software a mudanças no ambiente, nos requisitos e especificações funcionais. A Manutenibilidade possui cinco subcaracterísticas, são elas: Modularidade, Reusabilidade, Analisabilidade, Modificabilidade e Testabilidade.

A Portabilidade é o grau de eficiência e eficácia que um sistema, produto ou componente pode ser transferido de um ambiente de hardware, software ou outros ambientes de uso e operação, além do grau

em que o produto pode substituir com eficiência e eficácia outro produto similar. A Portabilidade possui três subcaracterísticas, são elas: Adaptabilidade, Instabilidade e Substitutibilidade.

2.2 SCRUM

O Scrum é uma das metodologias de gerenciamento de projetos originadas do movimento ágil, motivado pela necessidade de evolução das técnicas utilizadas até o momento, que teve como importante marco a assinatura do Manifesto Ágil, no qual Ken Schwaber e Jeff Sutherland, considerados “pais” do framework, estavam presentes (AGILE MANIFESTO, 2001).

Baseado em uma ideologia de desenvolvimento conhecida como iterativa e incremental, visa potencializar o resultado e a geração de valor em diversos tipos de projetos. De acordo com Schwaber et al (2017), o Scrum é uma metodologia que proporciona soluções criativas e adaptativas para problemas complexos, contudo, não define processos e métodos definitivos e imutáveis. Neste sentido, o Scrum é considerado um framework de gerenciamento que não oferece respostas prontas para os problemas do projeto, mas sim, caminhos e técnicas que auxiliem no desenvolvimento de soluções criativas e eficazes.

Desde sua formalização, por Schwaber (1997), o Scrum vem sendo aplicado com sucesso em projetos de áreas diversas, tornando-se extremamente popular nos âmbitos corporativo e acadêmico (SUTHERLAND et al., 2007). Apesar de frequentemente empregado do domínio de desenvolvimento de software, também é utilizado em ambientes como política, marketing, desenvolvimento de produtos, como, por exemplo, carros autônomos, entre outros (SCHWABER et al., 2017).

De acordo com dados evidenciados por Scrum Alliance (2015), uma pesquisa

realizada com quase cinco mil entrevistados, de países, empresas e áreas de atuação distintas, 82% dos entrevistados utilizam Scrum. Srivastava et al. (2017) apontam como principal motivo a possibilidade de iniciar um projeto sem a exigência de uma extensa fase de planejamento prévio, tal como sua capacidade de atender diversos tipos de projetos. Além disso, dados da pesquisa permitem concluir que o Scrum proporciona atratividade ao cliente, atendimento às restrições de escopo, orçamento e prazo e aumento na qualidade de trabalho.

Neste sentido, o Scrum não é apenas uma metodologia de desenvolvimento, mas sim, um framework que se mostrou atrativo para o desenvolvimento de produtos em áreas diversas.

Analisando suas principais características, um dos conceitos fundamentais do Scrum é sua abordagem iterativa-incremental que propõe que, a cada Sprint, um incremento do produto final seja gerado. De acordo com Hossain et al (2009), tal abordagem permite que o Scrum aplique a inspeção e adaptação, uma vez que o cliente tem a possibilidade de avaliar o incremento produzido e fornecer um feedback mais consistente à equipe.

Outra característica do Scrum é o tamanho reduzido de suas equipes que, de acordo com Schwaber et al (2017), deve variar entre 3 e 9 pessoas, além de ser autogerenciada, de forma que cada integrante deve conhecer suas responsabilidades e tem liberdade escolher a melhor forma de cumpri-las, o que, de acordo com Rising et al (2000), torna-a mais efetiva.

Além disso, muitas outras características podem ser atribuídas ao Scrum, algumas herdadas dos fundamentos ágeis, outras baseadas no esporte que deu origem a seu nome, o Rugby, outras próprias do framework. Schwaber (1997) cita algumas

delas, como, escopo flexível, colaboração entre os integrantes, revisão frequente do projeto como um todo, entre outras.

O Scrum prevê uma série de componentes que formam sua estrutura. De acordo com Schwaber et al (2017), cada componente tem relações com os demais, servem a um propósito específico e são fundamentais para o uso do Scrum. Esses componentes podem ser fragmentados conceitualmente em três grupos: papéis, eventos e artefatos.

Os papéis são funções que são distribuídas entre os integrantes da equipe Scrum, sendo eles, *Product Owner*, *Scrum Master*, *Development Team*. Cada um deles tem responsabilidades específicas no projeto (SUTHERLAND et al., 2007; SCHWABER et al, 2017; MOUNTAIN GOAT SOFTWARE, acessado em julho de 2020).

Os artefatos Scrum são ferramentas utilizadas como auxiliares na organização e desenvolvimento do produto, além de intensificarem a transparência, inspeção e adaptação do projeto. Os artefatos previstos por Schwaber et al (2017) são: *Product Backlog*, *Sprint Backlog* e incremento.

Os eventos, ou reuniões, do Scrum ocorrem ciclicamente até a entrega do produto final e tem como objetivo evitar que encontros não esperados sejam realizados, além de proporcionar momentos de inspeção e adaptação constantes no projeto. Todos os eventos possuem duração máxima, evitando que seja empregado mais esforço do que o necessário, e podem ser finalizados assim que seu objetivo for concluído. Os eventos são: *Sprint*, *Sprint Planning*, *Daily Meeting*, *Sprint Review* e *Sprint Retrospective* (RISING et al, 2000; SCHWABER et al, 2017; MOUNTAIN GOAT SOFTWARE, 2020).

Dadas as informações apresentadas até aqui, pode-se concluir que o Scrum é um framework de desenvolvimento ágil que pode potencializar projetos, contudo, deve

se observar que tais particularidades provém vantagens e desvantagens.

Uma de suas vantagens é citada por Hu Zhigen et al (2009), que afirmam que um dos benefícios oferecidos pelo Scrum é sua fácil implementação. Somando a isso, Hossain et al (2011) apresentam o aumento de comunicação, coordenação e controle do projeto. Já Mahalakshmi et al (2013), cita desvantagens como o baixo volume de documentação e a necessidade de uma equipe cooperativa e engajada. Apoiado por Meireles e Bonifácio (2015) afirmando que permite maior comunicação e colaboração entre os envolvidos no processo de desenvolvimento.

Pode-se afirmar, portanto, que o Scrum oferece diversas vantagens, de fato, contudo, algumas limitações e desvantagens também devem ser consideradas ao adotá-lo como metodologia.

Por fim, analisando da perspectiva de qualidade de produto de software, é possível verificar características vantajosas presentes no Scrum, principalmente considerando o MQPS definido pela ISO 25010. De acordo com o portal ISO 25000, qualidade de software é o grau em que o software atende a necessidade de suas partes, agregando valor ao produto. Neste sentido, verifica-se um alinhamento entre o conceito de qualidade definido pela ISO 25000 e o principal objetivo do framework, a geração de valor. Desta forma, é possível concluir que as características presentes no Scrum que possibilitam a entrega de valor podem afetar diretamente o grau de qualidade do produto na visão do modelo ISO 25010.

3. MATERIAIS E MÉTODOS

O presente artigo consiste em uma análise comparativa que faz uso de dados adquiridos através de questionários e entrevistas aplicadas a equipes de uma

organização real do mercado de desenvolvimento de software. O objetivo da análise é verificar possíveis relações entre a adoção do Scrum e o MQPS da ISO 25010. Para uma análise mais organizada, os resultados serão discutidos por pilar do modelo e os participantes serão divididos em perspectivas “Scrum”, representada pelos participantes que utilizam Scrum, e “Não Scrum”, representada pelos participantes que não utilizam.

3.1 OBJETO DE ESTUDO

Como objetos de estudo para o presente artigo foram adotadas equipes de desenvolvimento de uma organização real atuante do mercado de desenvolvimento de sistemas para automação de processos financeiros, no qual ela se mostra experiente e consolidada.

O que a organização em questão oferece a seus clientes pode ser segmentada em desenvolvimento de novos produtos e manutenção/suporte nos produtos já existentes (muitos considerados legados). Apesar do grande volume de produtos e serviços é visível o alto grau de padronização presente entre eles, o que pode ser atribuído a boa definição da estrutura de processos e componentes oferecida pela organização.

Em relação às metodologias, é possível verificar que equipes destinadas ao atendimento de produtos já existentes costumam manter-se utilizando metodologias tradicionais, enquanto as responsáveis por desenvolver novos produtos vêm optando por metodologias ágeis, como o Scrum.

Em relação às equipes selecionadas para a realização da análise comparativa, foram consideradas, inicialmente, duas equipes. A equipe A é formada por 12 integrantes e utiliza o Scrum para de o desenvolvimento de um novo produto. Já a equipe B é formada por 14 integrantes e utiliza de

metodologias tradicionais para dar suporte e manutenção a produtos já oferecidos pela organização. Considerando que as equipes adotadas como objeto de estudo do artigo são equipes reais do mercado de desenvolvimento, é necessário, para realização de análises consistentes, considerar que elas possuem objetivos e características próprias que podem influenciar nos resultados.

3.2 FONTES PRIMÁRIAS DE DADOS

Com o objetivo de realizar a comparação já citada, foram utilizados resultados providos por fontes primárias de dados, questionários e entrevistas aplicados aos objetos de estudo descrito.

O questionário em questão foi formulado em caráter transversal e aplicado à uma amostragem não probabilística intencional (os integrantes das equipes adotadas como objeto de estudo) com o objetivo de obter informações sobre a percepção dos participantes em relação ao grau de qualidade dos sistemas em que atuam quando submetidos aos pilares do MQPS da ISO 25010. O questionário é formado por perguntas fechadas, subdivididas em oito seções (uma para cada pilar do modelo de qualidade do produto, com uma pergunta para cada subcaracterística do pilar), além de uma seção inicial com duas perguntas de caráter pessoal e uma área de comentários livres ao final. Os resultados obtidos foram organizados e analisados de forma quantitativa.

As entrevistas foram desenvolvidas em formato estruturado utilizando amostragem não probabilística intencional (um líder de cada uma das equipes selecionadas), com objetivo de obter informações de caráter qualitativo sobre os processos utilizados por cada equipe em relação à qualidade de software do modelo da ISO 25010. A entrevista possui 18 perguntas abertas sobre a experiência profissional do entrevistado e

sobre o tratamento dos pilares do modelo em sua equipe.

Os questionários foram divulgados a todos os integrantes das equipes, contudo, houve poucos participantes da equipe B, que formaria a perspectiva “Não Scrum”, portanto, se mostrou necessária a adoção da equipe C, que possui objetivos e características semelhantes, possibilitando que seus resultados sejam compilados em uma mesma perspectiva. Desta forma, é possível estabelecer uma comparação balanceada entre os resultados providos pelas perspectivas “Scrum” e “Não Scrum”. As entrevistas foram aplicadas a ambos os líderes selecionados, um representante de cada perspectiva, por meio de videoconferência e transcritas para utilização no presente estudo.

4. RESULTADOS E DISCUSSÃO

A seção de Resultados e Discussão trará hipóteses sobre a relação entre o Scrum e cada pilar do MQPS da ISO 25010, seguidas da exposição e análise dos resultados obtidos através das fontes primárias.

4.1 ADEQUAÇÃO FUNCIONAL

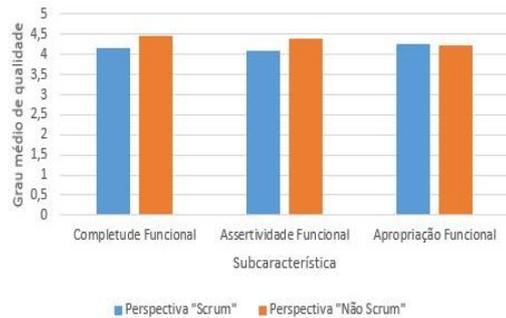
Sobre o pilar Adequação Funcional, a hipótese inicial apontava uma vantagem para a perspectiva “Scrum” por conta de suas características de desenvolvimento iterativo-incremental e feedback do cliente, que tornaria a avaliação das funcionalidades mais eficaz.

Os resultados, contudo, foram diferentes do esperado, como pode ser visualizado na Figura 1.

No gráfico apresentado pode-se verificar uma sutil vantagem da perspectiva “Não Scrum” nas duas primeiras subcaracterísticas e resultados quase idênticos na terceira. Analisando os dados obtidos por meio das entrevistas, pode-se

verificar que o processo de definição e validação de funcionalidades em ambas as perspectivas são similares, possivelmente por conta da padronização oferecida pela organização. Nesse sentido, tal processo aparenta ser mais impactante no pilar do que a utilização do Scrum.

Figura 1 - Gráfico de Adequação Funcional



Fonte: Os autores (2020)

4.2 EFICIÊNCIA DE DESEMPENHO

Sobre o pilar Eficiência de Desempenho, a hipótese inicial aponta que o Scrum não interferiria, mas sim, a arquitetura dos sistemas. Sendo assim, dada a padronização fornecida pela organização, os resultados deveriam ser semelhantes.

Como pode ser visto na Figura 2, as perspectivas demonstraram resultados consideravelmente diferentes.

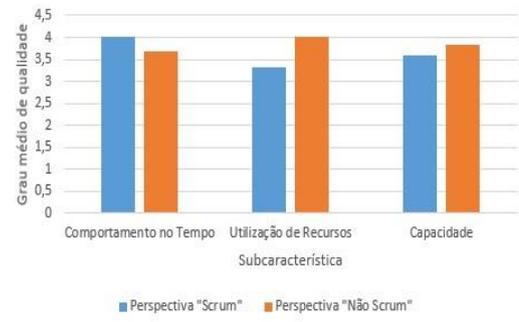
Sobre Comportamento no Tempo, a perspectiva "Scrum" demonstrou melhores resultados. Analisando as entrevistas, foi verificada uma preocupação com a avaliação do tempo de resposta das funcionalidades na perspectiva "Scrum", que não foi demonstrado pela perspectiva "Não Scrum", o que parece ser a justificativa para esta diferença.

Sobre as duas outras subcaracterísticas, não foram encontradas informações nas entrevistas que justificassem a diferença.

Neste sentido, é fato que existem diferenças entre as perspectivas, mas não aparentam ter relação com a utilização do Scrum, mas sim, com outros fatores, por exemplo, a

avaliação de desempenho realizada por cada equipe.

Figura 2 - Gráfico de Eficiência de Desempenho



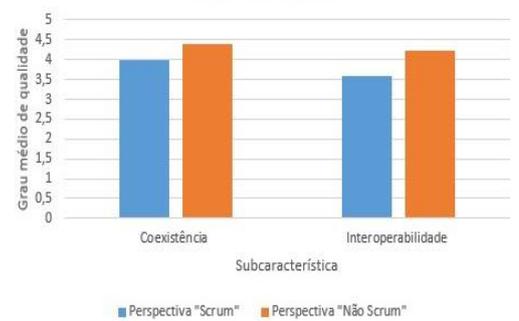
Fonte: Os autores (2020)

4.3 COMPATIBILIDADE

Sobre Compatibilidade, a hipótese inicial é de que o Scrum não apresenta características influentes neste pilar. Desta forma, o padrão oferecido pela organização faria com que as perspectivas apresentassem resultados semelhantes.

Apesar da hipótese apresentada, verifica-se na Figura 3 uma vantagem para a perspectiva "Não Scrum". Dado que a metodologia Scrum não possui diretrizes sobre compatibilidade, supõe-se que a vantagem é devida a outros fatores, como, possivelmente, a necessidade de cada produto.

Figura 3 - Gráfico de Compatibilidade



Fonte: Os autores (2020)

4.4 USABILIDADE

A hipótese sobre Usabilidade é segmentada. Para as subcaracterísticas Adequação em Familiaridade e Aprendizagem, a

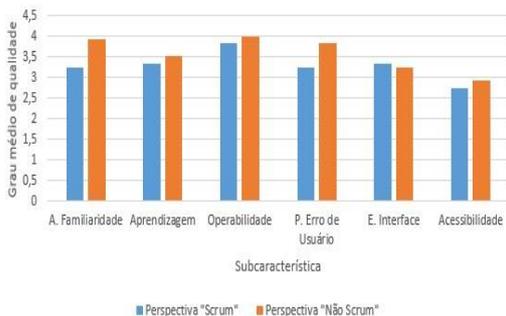
perspectiva “Scrum” teria desvantagens por conta do baixo grau de documentação em seus processos, uma vez que a documentação adequada seria crítica nestes pontos.

Em contrapartida, existiria uma possível vantagem nas subcaracterísticas Operabilidade e Proteção a Erro de Usuário devido à proximidade com o cliente e sua abordagem iterativa-incremental.

Por fim, as subcaracterísticas Estética da Interface de Usuário e Acessibilidade não se relacionam ao Scrum, pois a organização em questão utiliza uma padronização para estes pontos, o que tornaria os resultados similares.

Em relação às duas primeiras subcaracterísticas, conforme exposto na Figura 4, a hipótese se mostrou coerente.

Figura 4 - Gráfico de Usabilidade



Fonte: Os autores (2020)

Sobre Operabilidade e Proteção a Erro de Usuário, os resultados se mostraram diferentes do esperado, ou seja, vantajosos à perspectiva “Não Scrum”. Nas entrevistas foi verificado que tal perspectiva demonstrou um processo mais bem definido de avaliação de Usabilidade, o que aparenta ser o motivo da vantagem.

A hipótese de que não haveria diferenças nas subcaracterísticas de Estética de Interface e Acessibilidade, dado a padronização utilizada pela organização, diverge dos resultados obtidos.

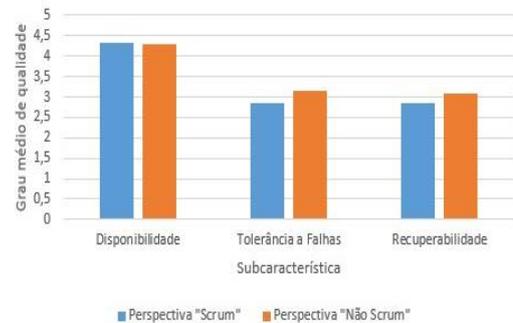
Com isso, é possível afirmar que as hipóteses se mostram parcialmente

coerentes, contudo, é viável supor que, além do uso do Scrum, outros fatores se mostraram influentes neste pilar, como, por exemplo, a robustez dos processos de avaliação de usabilidade

4.5 CONFIABILIDADE

Sobre Confiabilidade, mais uma vez é esperado que o Scrum não seja interferente, uma vez que não apresenta qualquer diretriz sobre este pilar.

Figura 5 - Gráfico de Confiabilidade



Fonte: Os autores (2020)

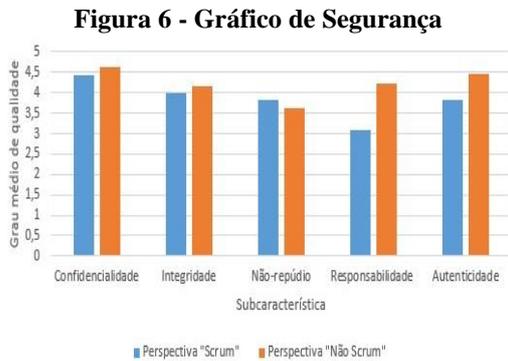
A Figura 5 mostra resultados que divergem da hipótese, onde duas das três subcaracterísticas demonstram uma sutil, porém existente, vantagem para a perspectiva “Não Scrum”. Analisando as entrevistas realizadas não foram encontradas informações que justificassem essa divergência, contudo, provavelmente, não é atribuída ao Scrum, mas sim, a outros fatores.

4.6 SEGURANÇA

Em relação ao pilar Segurança, também é esperado que o Scrum não interfira por não mostrar características em relação a este ponto. As equipes estudadas utilizam a mesma estratégia de segurança padronizada pela organização, o que, teoricamente, resultaria em uma semelhança nas respostas do questionário.

Na Figura 6 é possível verificar que os resultados, diferente do que era esperado,

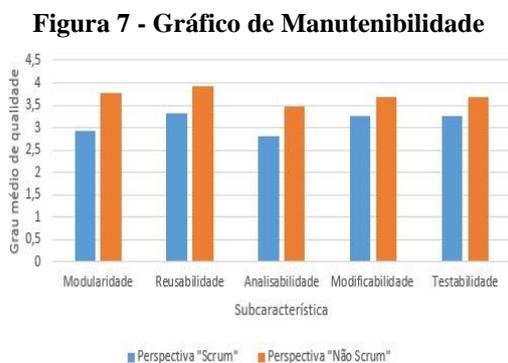
mostraram distanciamento entre as perspectivas. Não foram encontradas informações nas entrevistas que justifiquem a diferença encontrada, contudo, a suposição é que tal distanciamento tem relação com a necessidade de cada produto e o uso das estratégias de segurança.



Fonte: Os autores (2020)

4.7 MANUTENIBILIDADE

Sobre Manutenibilidade, a hipótese é de que, falando do pilar Modularidade, as perspectivas demonstraram resultados similares, uma vez que este fator seria afetado diretamente apenas pela estrutura dos sistemas, onde existe uma padronização por parte da organização. Em relação às demais subcaracterísticas, a suposição é de que o Scrum impacta negativamente, uma vez que não se preocupa tanto com a documentação do produto, o que pode ser essencial nestes pontos.



Fonte: Os autores (2020)

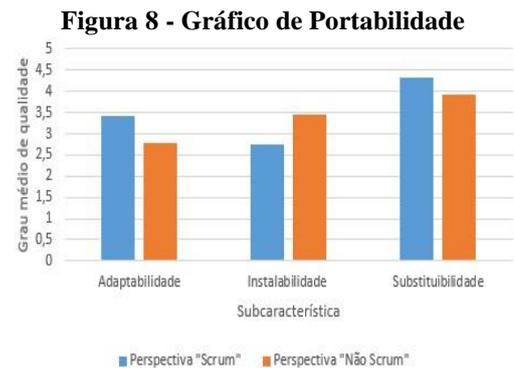
Analisando a Figura 7 é possível verificar que, sobre a modularidade, diferente da

hipótese, houve uma vantagem para a perspectiva “Não Scrum”. Neste sentido, deve-se considerar que, além da estrutura do sistema, outros fatores podem ser interferentes.

A hipótese levantada sobre as demais subcaracterísticas se mostrou coerente com os resultados obtidos.

4.8 PORTABILIDADE

Em relação ao pilar Portabilidade, a hipótese levantada é de que a metodologia de desenvolvimento utilizada não interfere nos níveis de qualidade. As tecnologias utilizadas, a estrutura do sistema, robustez, entre outros fatores, são propícios a influenciarem o quesito citado.



Fonte: Os autores (2020)

Conforme pode ser verificado na Figura 8, houve distanciamentos entre as perspectivas em todas as subcaracterísticas, entretanto, não foram encontradas informações nas entrevistas que os justificassem. É viável considerar que existem fatores que influenciaram em tal divergência, entretanto, nada aponta que eles tenham relação com o Scrum.

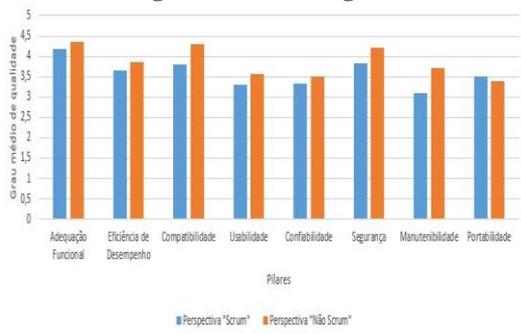
4.9 ANÁLISE GERAL

Os resultados gerais podem ser vistos na Figura 9.

Apesar da maioria das hipóteses apontarem que o Scrum não seria interferente, é

possível verificar que para a maioria dos pilares, com exceção de Portabilidade, a perspectiva “Não Scrum” obteve melhores graus de qualidade. Além disso, poucos foram os resultados que não mostraram significativo distanciamento entre as perspectivas.

Figura 9 - Gráfico geral



Fonte: Os autores (2020)

Sobre as hipóteses levantadas, de forma geral, algumas se mostraram coerentes, outras não. Para as hipóteses que divergiram dos resultados, ora foi possível encontrar justificativas e formular novas hipóteses baseadas em informações adquiridas, principalmente, através das entrevistas realizadas, contudo, houve resultados que ficaram inconclusivos.

Vale lembrar que as equipes utilizadas como objeto de estudo estão inseridas em um ambiente real de desenvolvimento de software, portanto, é necessário considerar que diversas características podem ser influenciadoras dos resultados, por exemplo, a experiência dos integrantes, o objetivo das equipes e o domínio de seus produtos.

5. CONCLUSÃO

Dado o aumento na preocupação com a qualidade de software por parte das organizações de desenvolvimento e, em paralelo, a crescente adoção do Scrum como alternativa de metodologia de desenvolvimento, verifica-se a importância do presente estudo, uma vez que busca

identificar possíveis relações entre as abordagens citadas em um ambiente real do mercado.

Os objetivos específicos citados na seção introdutória foram realizados no decorrer do artigo, portanto, considera-se concluído o objetivo geral, apresentado na mesma seção.

Sobre as análises, diversos resultados foram apresentados e discutidos, contudo, alguns devem ser destacados. Um deles é que, apesar das hipóteses sobre os pilares, em maioria, apontaram que o Scrum não seria interferente, em sete dos oito pilares a perspectiva “Não Scrum” mostrou melhores graus de qualidade. Além disso, apesar desta hipótese, raras foram as questões que não demonstraram certo distanciamento entre as perspectivas em seus resultados.

Outro resultado relevante, desta vez diretamente relacionado ao Scrum, é que, aparentemente, a característica de baixa documentação da metodologia é prejudicial a alguns pontos de qualidade dos pilares Manutenibilidade e Usabilidade.

5.1 SUGESTÕES DE TRABALHO FUTUROS

Considerando que o presente estudo é realizado utilizando equipes reais do mercado de desenvolvimento, torna-se interessante, como trabalho futuro, aplicar uma análise semelhante em outras equipes, com características diferentes, e avaliar os resultados.

Outra sugestão interessante seria de realizar um estudo semelhante, contudo, aplicando avaliações e métricas de qualidade aos produtos, além de considerar a opinião dos integrantes das equipes.

Além disso, se mostra promissora a variação do estudo, utilizando outras metodologias de desenvolvimento e modelos de qualidade.



AGRADECIMENTOS

Agradecemos, a Deus por nos proporcionar bênção e luz para nossas vidas.

A Faculdade de Tecnologia da Zona Leste, por oferecer estrutura para que pudéssemos realizar este trabalho.

Ao nosso orientador Luciano Francisco de Oliveira, por nos assessorar.

A organização objeto de estudo, por acreditar em nosso trabalho e colaborar ativamente para que esse estudo acontecesse.

Aos líderes das equipes, que tiveram contribuição fundamental para o sucesso da pesquisa.

Aos que responderam os questionários, por ajudarem no levantamento de informações.

Ao professor Amilton Quintela e demais colaboradores do curso "TCC Sem Drama", por auxiliar no desenvolvimento do trabalho.

REFERÊNCIAS

AGILE MANIFESTO. **History: The Agile Manifesto**. AGILE MANIFESTO. 2001. Disponível em: <https://agilemanifesto.org/history.html>. Acesso em: 2 jul. 2020.

BARTIÉ, Alexandre. **Garantia da qualidade de software: adquirindo maturidade organizacional** / 13. Ed. Rio de Janeiro: Elsevier, 2002.

COLLINS, Eliane F.; LOBÃO, Luana M. de A. **Experiência em Automação do Processo de Testes em Ambiente Ágil com SCRUM e ferramentas OpenSource**. In: PROCEEDINGS OF IX SIMPÓSIO BRASILEIRO DE QUALIDADE DE SOFTWARE. 2010.

GUERRA, Ana C. G; COLOMBO, Regina M. T. **Tecnologia da Informação: Qualidade de Produto de Software**. MCT/SEPIN, Brasília, p. 429, 2009.

HOSSAIN, Emam; BABAR, Muhammad Ali; PAIK, Hye-young. **Using scrum in global**

software development: a systematic literature review. In: 2009 FOURTH IEEE INTERNATIONAL CONFERENCE ON GLOBAL SOFTWARE ENGINEERING. IEEE, 2009. 175-184 p.

HOSSAIN, Emam; BANNERMAN, Paul L.; JEFFERY, D. Ross. **Scrum practices in global software development: a research framework**. In: INTERNATIONAL CONFERENCE ON PRODUCT FOCUSED SOFTWARE PROCESS IMPROVEMENT, Berlin, Heidelberg: Springer, 2011. 88-102 p.

MAUDA, Everson C. **Modelo de qualidade para características internas de segurança de componentes de software**. Dissertação (mestrado) – Pontifícia Universidade Católica do Paraná: Curitiba, 2012

SAMBO, José Luís. **Garantia de qualidade de software In.: INFORMÁTICA APLICADA: CSI 5302**. Universidade Virtual Africana, pp.17-19. avu.org, 2018.

HU, Zhi-gen; YUAN, Quan; ZHANG, Xi. **Research on agile project management with scrum method**. In: 2009 IITA INTERNATIONAL CONFERENCE ON SERVICES SCIENCE, MANAGEMENT AND

ENGINEERING. IEEE, 2009. 26-29 p.

ISO/IEC 25010. **ISO/IEC 25010:2011: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuARE) — System and software quality models**. 2011.

ISO 25000. **ISO/IEC 25010**. ISO 25000. Disponível em: <https://iso25000.com/index.php/en/iso-25000standards/iso-25010>. Acesso em: 20 nov. 2020.

MAHALAKSHMI, M.; SUNDARARAJAN, M. **Traditional SDLC Vs Scrum Methodology—A Comparative Study**. International Journal of Emerging Technology and Advanced Engineering, p. 192-196, 2013.

MEIRELES, Maria A. C.; BONIFÁCIO, Bruno A. **Uso de Métodos Ágeis e Aprendizagem Baseada em Problema no Ensino de Engenharia de Software: Um Relato de Experiência**. In.: Anais do XXVI Simpósio



Brasileiro de Informática na Educação (SBIE). CBIE-LACLO, 2015.

MOUNTAIN GOAT SOFTWARE. **Scrum Overview for Agile Software Development**. MOUNTAIN GOAT SOFTWARE. Disponível em:

<https://www.mountaingoatsoftware.com/agile/scrum/resources/overview>. Acesso em: 18 nov. 2020.

PRESSMAN, R. S. **Software Engineering: A Practitioner's Approach**. 7. ed. McGraw Hill, 2010.

RISING, Linda; JANOFF, Norman S. **The Scrum software development process for small teams**. IEEE software, v. 17, n. 4, p. 26-32, 2000.

SANDERS, Joc; CURRAN, Eugene. **Software quality: a framework for success in software development and support**. USA: ACM Press/Addison-Wesley Publishing Co, 1994.

SCHWABER, Ken. **Agile project management with Scrum**. Microsoft press, 2004.

SCHWABER, Ken. **Scrum development process**. In: BUSINESS OBJECT DESIGN AND

IMPLEMENTATION, London: Springer, 1997. 117-134 p.

SCHWABER, Ken; SUTHERLAND, Jeff. Um guia definitivo para o Scrum: **As regras do jogo**. Scru Org. Tradução Fábio Cruz & Eduardo Rodrigues Sucena. 2017. Tradução de: The Definitive Guide to Scrum.

Disponível em:

<https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-PortugueseBrazilian.pdf>. Acesso em: 18 nov. 2020.

SCRUM ALLIANCE. **The 2015 State of Scrum Report**. Scrum Alliance. 2015. Disponível em:

<https://www.scrumalliance.org/ScrumRedesignDEVSite/media/scrumalliancemedi/files%20and%20pdfs/state%20of%20scrum/scrum-alliance-state-of-scrum-2015.pdf>. Acesso em: 18 nov. 2020.

SRIVASTAVA, Apoorva; BHARDWAJ, Sukriti; SARASWAT, Shipra. **SCRUM model**

for agile methodology. In: 2017 INTERNATIONAL CONFERENCE ON COMPUTING, COMMUNICATION AND AUTOMATION (ICCCA). IEEE, 2017. 864-869 p.

SUTHERLAND, Jeff; SCHWABER, Ken. **The Scrum Papers: Nuts, Bolts and Origins of an Agile Process**. 2007.

TSUKUMO, Alfredo N. et al. **Qualidade de Software: Visões de Produto e Processo de Software**. In: II Escola Regional de Informática da Sociedade Brasileira de Computação Regional de São Paulo - II ERI da SBC. 1997.