

**AS DIFICULDADES DE GERENCIAMENTO DE UM TIME SCRUM ACADÊMICO
PARA O DESENVOLVIMENTO DE UM SOFTWARE PARA ENTIDADES
FILANTRÓPICAS**

***THE MANAGEMENT DIFFICULTIES OF AN ACADEMIC TIME SCRUM FOR THE
DEVELOPMENT OF A SOFTWARE FOR PHILANTHROPIC ENTITIES***

Pedro Cassone Rodrigues de Campos – pedrocassone@gmail.com

Daniela Gibertoni – daniela.gibertoni@fatectq.edu.br

Faculdade de Tecnologia de Taquaritinga (FATEC) – SP – Brasil

DOI: 10.31510/infa.v15i2.497

RESUMO

O desenvolvimento de produtos *software* sem o auxílio de uma metodologia específica, pode vir a acarretar falta de qualidade e problemas ao decorrer do projeto. Para tanto, este artigo apresenta um estudo de caso que visa investigar as principais dificuldades de gestão de uma equipe Scrum para o desenvolvimento de um sistema que faz doações para entidades filantrópicas. Este estudo de caso foi realizado pelo Grupo de Pesquisa em Engenharia de Software – GPES da Faculdade de Tecnologia de Taquaritinga em parceria com uma empresa de desenvolvimento. Conseguiu-se concluir que as metodologias ágeis pouco têm de novo, porém, quando se diz respeito ao seu enfoque, consegue-se notar que ele muda totalmente, de processos para as pessoas. As metodologias ágeis não mudam somente o enfoque, mas também mudam a maneira de como se passa a enxergar os processos envolvidos em projetos, uma observação final sobre a utilização do Scrum no GPES.

Palavras-chave: Engenharia de Software. Desenvolvimento ágil. Scrum. Cultura ágil.

ABSTRACT

The development of software products without the aid of a specific methodology may lead to a lack of quality and problems in the course of the project. To do so, this article presents a case study that aims to investigate the main difficulties of managing a Scrum team for the development of a system that makes donations for philanthropic entities. This case study was carried out by the Software Engineering Research Group - GPES of Taquaritinga's Faculty of Technology in partnership with a development company. It has been concluded that agile methodologies are little new, but when it comes to their focus, it can be noticed that it changes completely, from processes to people. Agile methodologies not only change the focus, but also change the way in which the processes involved in projects are viewed, a final observation about the use of Scrum in the GPES.

Keywords: Software Engineering. Agile development. Scrum. Agile culture.

1 INTRODUÇÃO

Quando se inicia um projeto, seja ele qual for, precisa-se antes definir um norte, estipular prazos e objetivos a curto, médio e longo prazo, com um projeto de *software* também é assim. Entender os objetivos, definir metas e gerenciar pessoas não é um trabalho simples, o gerenciamento consiste em adequar cada recurso que se possui da melhor maneira possível, a fim de conseguir criar agilidade e produtividade dentro de um cenário com prazos cada vez mais curtos e com requisitos que mudam constantemente.

Neste cenário, desenvolver um *software* com qualidade, que consiga atender a todos os requisitos exigidos pelo cliente em pouco tempo e com poucos desenvolvedores parece algo inatingível, porém, com o apoio de metodologias ágeis e com definições claras de papéis em seu time de desenvolvimento, essa missão se torna mais real e factível.

O desenvolvimento de um *software* sem o auxílio de uma metodologia específica pode vir a resultar em um produto sem a qualidade esperada. Porém, a utilização de metodologias clássicas pode fazer com que projetos que não necessitem de todo o formalismo existente em sua ideia, acabe ficando travado em seu desenvolvimento e não flua de uma maneira simples e rápida. Em outras palavras, não flua de uma maneira ágil (SBROCCO; MACEDO, 2012).

Este artigo tem como objetivo realizar um estudo de caso que visa investigar as principais dificuldades de gestão de um time Scrum acadêmico para o desenvolvimento de um *software* que faz doações para entidades filantrópicas, apresentando para isso, os indicadores de produtividade do projeto.

O presente trabalho está organizado em cinco seções: a seção 2 apresenta a metodologia de pesquisa, a seção 3 a fundamentação teórica a respeito do Scrum, tendo como sequência na seção 4 uma breve discussão sobre Desenvolvimento Distribuído de Software (D.D.S). A seção 5 apresenta todo o estudo de caso desenvolvido bem como os resultados encontrados. Na seção 6 é explanado sobre as conclusões referentes a este trabalho.

2 METODOLOGIA DE PESQUISA

Na elaboração deste artigo foi necessário a utilização da pesquisa bibliográfica, que é constituída por buscas através de dados obtidos em livros, ou artigos científicos e caracteriza-se como pesquisa exploratória (GIL, 2008). Além da pesquisa bibliográfica, utilizou-se ainda

o estudo de caso que na visão de Gil (2008, p.58), “é caracterizado pelo estudo profundo e exaustivo de um ou de poucos objetos, de maneira a permitir o seu conhecimento amplo e detalhado, tarefa praticamente impossível mediante os outros tipos de delineamentos considerados”.

3 METODOLOGIA ÁGIL – SCRUM

Segundo Pressman (2011), o Scrum é um modelo ágil de processo de desenvolvimento que foi desenvolvido por Jeff Sutherland e por sua equipe no início da década de 1990. Na sua origem, o Scrum foi desenvolvido para ser implementado em equipes de desenvolvimento de produtos de *software*. Com o passar do tempo, passou a ser utilizado por qualquer empresa que necessite implementar processos de gerenciamento de projetos, tais como agências de publicidade ou projetos de arquitetura.

O modelo de equipe no Scrum é projetado para otimizar a flexibilidade, criatividade e produtividade, contendo três papéis principais: o Scrum *Master*, o *Product Owner* e o *Team*. Eles devem ser auto organizados, pois são responsáveis por escolher a melhor forma de realizar um trabalho (SCHWABER; SUTHERLAND, 2013).

- a) *Scrum Master*: É o responsável por garantir que o Scrum seja compreendido e implantado com sucesso. O *Scrum Master* tem como por objetivo garantir que o *Team* adere à teoria, práticas e regras do Scrum de maneira satisfatória (SCHWABER; SUTHERLAND, 2013);
- b) *Product Owner*: Também conhecido como dono do produto, tem como função, maximizar o valor do produto e do trabalho do *Team*, ele é responsável por coletar e escrever os requisitos (estórias) que estarão inseridos no *backlog*, além de gerenciar e manter as mesmas (SCHWABER; SUTHERLAND, 2013);
- c) *Team*: É definido como o time de desenvolvimento e é constituído por profissionais que realizam o trabalho de desenvolver e entregar uma versão funcional (SCHWABER; SUTHERLAND, 2013).

Para um bom entendimento do Scrum faz-se necessário falar sobre os pilares dele, assunto do próximo item.

3.1 Pilares

O Scrum é fundamentado em três pilares que auxiliam e apoiam seu processo de controle empírico, são eles: transparência, inspeção e adaptação (SCHWABER; SUTHERLAND, 2013).

- a) **Transparência:** Garante que os aspectos referentes ao processo devem estar disponíveis aos envolvidos nos resultados. Requer que tais processos sejam regidos por um padrão, para que assim, todos os envolvidos no projeto estejam compartilhando o mesmo entendimento (SCHWABER; SUTHERLAND, 2013);
- b) **Inspeção:** Frequentemente deve haver inspeção nos artefatos Scrum e no progresso com o objetivo de detectar mudanças (SCHWABER; SUTHERLAND, 2013);
- c) **Adaptação:** Se for comprovado que algum aspecto de um processo sofreu um desvio que irá resultar em um produto resultado inaceitável, o produto ou processo deve ser adaptado imediatamente a fim de evitar mais desvios (SCHWABER; SUTHERLAND, 2013).

O Scrum tem como base ainda, os eventos definidos, que devem ser seguidos pelo time de desenvolvimento, esses eventos são abordados no item abaixo.

3.2 Eventos

Eventos prescritos são usados no Scrum para criar uma rotina e minimizar a necessidade de reuniões não definidas. Todos os eventos são eventos *time-boxed*, ou seja, todo evento tem uma duração máxima (SCHWABER; SUTHERLAND, 2013). Os principais eventos são:

- a) **Sprint:** O núcleo do Scrum é a *Sprint*, um *time-boxed* de um mês ou menos, durante o qual um “Pronto”, versão incremental potencialmente utilizável do produto, é criado. Sprints tem durações coerentes com todo o esforço de desenvolvimento (SCHWABER; SUTHERLAND, 2013).
- b) **Reuniões:** No Scrum, são definidas reuniões durante o processo de desenvolvimento que possuem um objetivo em específico e diferenciado para cada situação ou período no qual a mesma ocorre.

Os artefatos no Scrum, representam trabalho ou valor, para tanto, esse assunto será abordado no item abaixo.

3.3 Artefatos

Representam o trabalho ou o valor para que possa haver transparência, avaliação e adaptação (SCHWABER; SUTHERLAND, 2013). Os principais artefatos do Scrum são:

- a) *Backlog* do Produto: Consiste em uma lista que apresenta de maneira ordenada e priorizada, todos os requisitos (SCHWABER; SUTHERLAND, 2013);
- b) *Backlog* da Sprint: É um conjunto de itens do *backlog* do Produto selecionados e adicionados à lista de afazeres de uma Sprint. Representa uma previsão de quantas e quais funcionalidades o *Team* acredita alcançar durante aquele período (SCHWABER; SUTHERLAND, 2013).

O Scrum foi utilizado em junção com o Desenvolvimento Distribuído de Software (D.D.S), portanto, é apresentado no item a seguir uma breve explicação sobre ele.

4 DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE (D.D.S)

O processo de desenvolvimento de *software* não pode ser limitado e entendido como um processo presencial. Existe a possibilidade da união de pessoas em diferentes localidades com o objetivo de desenvolver em conjunto um *software*. Esse tipo de desenvolvimento é conhecido como Desenvolvimento Distribuído de Software (D.D.S) (SIQUEIRA, 2005).

D.D.S é um modelo onde os envolvidos em um determinado projeto estão dispersos. De acordo com Siqueira (2005), as vantagens que o D.D.S oferece são:

- a) Possibilidade de desenvolvimento *follow-the-sun*, ou seja, 24 horas contínuas de trabalho, que permite o aumento de produtividade e a redução dos prazos de entrega dos produtos;
- b) Disponibilidade de recursos globais com baixos custos e a qualquer hora;
- c) Disponibilidade de recursos qualificados em áreas especializadas;
- d) Possibilidade de formação de equipes virtuais.

A maior padronização e formalidade do time, causado pela menor familiaridade e proximidade entre os integrantes e a separação física das pessoas fazendo com que elas conversem menos informalmente pode ser definida como um dos benefícios apresentados pelo D.D.S (LAYZELL; BRERETON; FRENCH, 2000). Essa similaridade foi comprovada pelo grupo ao presenciar tais características ao decorrer do projeto. Portanto, foi elaborado um estudo de caso que descreve o desenvolvimento de um *software* em cooperação com uma empresa de desenvolvimento utilizando uma adaptação do Scrum juntamente com o D.D.S. Esse *software* gerencia doações para entidades filantrópicas e está descrito no estudo de caso a seguir.

5 ESTUDO DE CASO

O estudo de caso foi realizado pelo Grupo de Pesquisa em Engenharia de Software – GPES – da Faculdade de Tecnologia de Taquaritinga - Fatec. O GPES é um grupo que tem como objetivo realizar pesquisas na área de Engenharia de Software por meio de parcerias e/ou convênios com empresas e atua desde 2010, tendo desenvolvido vários projetos. Por sua vez, a Fatec Taquaritinga é uma das 71 instituições públicas de ensino superior do Centro Paula Souza no Estado de São Paulo. Há mais de 25 anos, sua missão é oferecer educação gratuita e de qualidade a alunos de Taquaritinga e região, formando tecnólogos em áreas privilegiadas no mercado de trabalho com seus cursos presenciais em Agronegócio, Análise e Desenvolvimento de Sistemas, Produção Industrial e Sistemas para Internet.

O projeto aqui apresentado tem como principal objetivo, a implementação de um *software* de gerenciamento de doações para entidades filantrópicas. A solução tem como principal foco automatizar e auxiliar as operadoras a arrecadarem doações através de ligações, além de oferecer uma ferramenta eficiente de gestão para as entidades, tendo como principal diferencial a plataforma de desenvolvimento *WEB*, o que além de permitir a integração, também permite a utilização de tecnologias atuais e responsivas. Portanto o presente projeto tem como principal foco e gênero a ação social e os processos de negócio presentes em entidades beneficentes. A solução proposta para esse projeto utiliza tecnologias atuais e que possuem grandes tendências futuras em relação a sua utilização. Nesse aspecto, caracteriza-se como tecnologia, o *framework* Angular e a linguagem C#, tendo a base projetada e implementada no *SQL Server Management Studio*. Para o desenvolvimento do projeto, a metodologia ágil Scrum foi escolhida, ela foi a opção de maior viabilidade devido às suas

características e seus benefícios analisados a curto, a médio e a longo prazo. O Scrum apresenta não somente a possibilidade de maior participação entre o cliente e o *team*, mas também apresenta possibilidades de trabalho descentralizado, o que para o grupo é uma necessidade de extrema importância. A metodologia também foi implantada a fim de se obter uma produtividade superior por parte do *team*, além de alcançar prazos menores o que é uma das principais exigências no desenvolvimento de produtos de *software* atualmente.

5.1 A utilização da metodologia SCRUM

O Scrum foi a metodologia de desenvolvimento proposta a ser utilizada no decorrer deste projeto, onde o GPES analisou vários fatores de sua estrutura, suas características e sua utilização atual, antes de sua escolha efetiva. No momento em que a metodologia foi selecionada, o requisito mínimo a ser cumprido por todos os envolvidos no projeto era a adaptação do pensamento clássico de desenvolvimento para um pensamento ágil. Nesse sentido, analisa-se um choque inicial do grupo no primeiro momento em que eles se depararam com a ideologia e com a implantação do Scrum. Entretanto, alterar um pensamento clássico de desenvolvimento para um padrão ágil, leva tempo, disposição, dedicação e sobretudo a aceitação. Esse momento de dúvida e desorientação por parte de membros do time de desenvolvimento, tende a ser superado com a presença efetiva de um Scrum *Master* no grupo, o mesmo foi importante no GPES, pois com ele houve uma preparação do cenário atual de desenvolvimento do grupo, além das realizações das adaptações necessárias, a fim de garantir a produtividade da metodologia.

A implantação do Scrum no grupo levou em consideração os membros, ou seja, a capacidade que cada membro do grupo tem em abstrair os ideais da metodologia. Inicialmente, a implantação ocorreu de forma dividida, ou seja, o Scrum não foi totalmente inserido logo no começo do projeto, pois, o conhecimento dos membros sobre o mesmo era relativamente baixo, o que além de prejudicar o seu fluxo, também atrapalharia seu entendimento. A cada etapa da implantação do Scrum, uma melhoria era analisada, não somente na adaptação da metodologia, mas também na forma do relacionamento e da gestão do time de desenvolvimento. Uma das etapas mais básicas e simples da metodologia Scrum, consiste na separação e atribuição de papéis aos membros do grupo, essa divisão consiste em repassar aos membros sua função principal que ele iria desempenhar no decorrer do projeto, sendo uma importante etapa. Identificar os papéis de cada membro ajudou inicialmente a

localização da situação do grupo e obviamente a identificar os recursos disponíveis à conclusão do projeto. Essa definição ocorreu de forma transparente, simples e clara para os membros, pois o importante é o mesmo não se perder. A observação que pode ser realizada a respeito dessa etapa no GPES, é a de que a mesma ocorreu de uma maneira transparente com todos os integrantes, respeitando as afinidades de cada um separadamente, ou seja, os integrantes passaram suas facilidades e as mesmas foram levadas em consideração no momento da atribuição das tarefas, ponto importante a ser destacado que prevê exatamente essa atenção e acompanhamento ao decorrer do projeto. Conseguir compreender sua função e qual deverá ser sua postura ao decorrer do projeto é um requisito do integrante do grupo, porém, apenas com o auxílio e orientação, esse objetivo é alcançado, especificamente ao primeiro momento de inexperiência, os membros se sentem mais seguros com o apoio da gestão, pois todos os papéis ainda são confusos ao meio, então, criar a cultura da auto gestão nesse ambiente é quase que impossível. Desta forma, a experiência descrita neste estudo de caso envolveu a participação de cinco integrantes e o *Product Owner*, a saber:

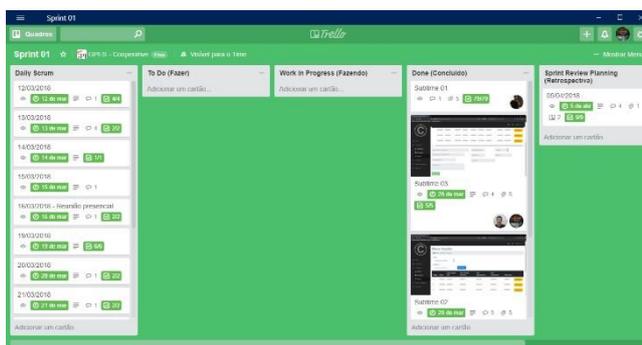
- a) Três integrantes desenvolvedores: responsáveis pela implementação das histórias;
- b) Um integrante responsável pela base de dados: responsável pelo projeto e implementação;
- c) Um integrante *Scrum Master*: responsável por garantir o sucesso do Scrum;
- d) O *Product Owner*: responsável por relatar as regras de negócio e criar o *backlog*.

Após a definição base de papéis, o próximo passo executado foi o levantamento de requisitos ou escrita de requisitos na metodologia ágil, mais conhecidos como histórias. Nesse estudo de caso, quem escreveu as histórias e, conseqüentemente, gerenciou o *Product Backlog* foi o *Scrum Master*. Esta observação é importante, dado que foi a primeira adaptação do Scrum junto ao GPES. Portanto, o papel do *Scrum Master* foi fundamental para que a evolução do projeto se concretizasse.

Com todo o *Product backlog* pronto e priorizado houve a necessidade da responsabilidade para com os eventos Scrum, o aspecto a ser frizado nesse ponto do desenvolvimento do projeto é a participação presencial de parte do time de desenvolvimento apenas uma vez por semana. O time de desenvolvimento trabalhava à distância, tendo a inspeção realizada da mesma maneira, todos os eventos Scrum precisaram ter recebido adaptações para uma espécie de “*home office development*”. Trabalhar a distância para o desenvolvimento do projeto exigiu a utilização de ferramentas, como por exemplo *GitHub* e

Trello. Estes *softwares* foram escolhidos por serem livres e foram utilizados a fim de amenizar o efeito que a ausência do time causava no projeto. Atenta-se ao *Trello*, ferramenta de gestão, que tem como principal objetivo o controle dos eventos Scrum dentro de um projeto, ela utiliza-se do sistema Kanban, e fora de suma importância para a gestão do não presencial do time. Está sendo demonstrada na Figura 1, a fim de comprovar sua utilização e efetividade na gestão do *team*:

Figura 1: Captura de tela da ferramenta Trello.



Fonte: Elaborado pelos autores (2018).

Os ciclos do Scrum que foram sendo completos eram sempre inspecionados pelo Scrum *Master* que no primeiro momento coincidiu como um gerente de projetos. Cada um dos ciclos manteve seu tempo fixado em duas semanas, tendo sempre em vista a disponibilidade dos membros do grupo com o projeto, reuniões eram realizadas através de mensagens enviadas em aplicativos sociais, a fim de se obter inspeção e controle do trabalho que estava sendo realizado. Ao final de cada Sprint reuniões de apresentações de incrementos com a presença do cliente eram realizadas e o grupo discutia o trabalho que fora desenvolvido, presencialmente era possível os encontros para essas discussões uma vez por semana. Ao final do replanejamento, o grupo apresentava sugestões e/ou reclamações a respeito da metodologia. Novas instruções eram repassadas aos mesmos, sobre os próximos passos do desenvolvimento do projeto e o ciclo se repetia. Para entender essa insatisfação foi realizado entrevistas individuais com cada integrante, de modo a avaliar qualitativamente e quantitativamente a efetividade da metodologia. É o que está descrito no próximo item.

5.2 Desempenho dos integrantes do grupo

O Scrum exige mudança, seja na maneira de pensar, seja na maneira de gerenciar. No primeiro momento de inserção de uma metodologia ágil no grupo, conseguiu-se observar claramente a resistência à metodologia por parte de alguns integrantes, eles apresentaram não somente o não entendimento, mas também o não interesse. Além de não participarem de maneira efetiva no grupo em um primeiro momento. A própria observação já demonstrava resultados negativos que seriam alcançados caso não houvesse uma gestão estratégica dos membros. Nesse momento inicial, os membros passaram por uma espécie de avaliação, em que os pontos fortes e fracos de cada um, foi exposto a fim de se compreender sua motivação, ideais e opiniões. Com uma linha de alcance de “felicidade” ou “satisfação” traçada, o gerenciamento iniciou tendo em vista e em objetivo a motivação dos membros a concluírem as Sprints e a trabalharem em conjunto. Após a avaliação inicial, os dados obtidos foram inseridos em uma tabela de acompanhamento, apresentada na Tabela 1, a qual utiliza-se de planilhas eletrônicas a fim de efetuar a medição de métricas alcançadas pelo *team* durante o trabalho.

Tabela 1: Dados recolhidos durante o projeto.

Tópicos	01	02
PONTOS RESOLVIDOS	50pts.	30pts.
FELICIDADE	2pts.	5pts.
PONTOS PLANEJADOS	50pts.	30pts.
VELOCIDADE PONTOS PLANEJADOS	100%	100%
META VELOCIDADE	100%	100%
PONTOS RESOLVIDOS E NÃO PLANEJADOS	25pts.	10pts.
MUDANÇA DE ESCOPO	35%	29%

META MÁXIMA/MUDANÇA	50%	50%
---------------------	-----	-----

Fonte: Elaborado pelos autores (2018).

A Tabela 1 possui os referidos tópicos que estão descritos e detalhados abaixo:

- a) Pontos resolvidos: dizem respeito ao total de pontos das estórias que o *team* conseguiu desenvolver durante a Sprint;
- b) Felicidade: nota de 0 até 5 que representa a satisfação média (bem-estar) dos membros do *team* por terem participado da Sprint, essa nota é atribuída secretamente;
- c) Pontos planejados: dizem respeito ao total de pontos das estórias que o *team* se propôs a concluir durante a Sprint;
- d) Velocidade da conclusão dos pontos planejados: consiste em uma relação dos pontos resolvidos por pontos planejados;
- e) Meta de velocidade de conclusão: manter nossa meta de velocidade em 100% ou próxima, pois este tópico relaciona a velocidade que o *team* deve alcançar, ou seja, a conclusão referente as estórias planejadas;
- f) Pontos resolvidos e não planejados: diz respeito a quantos pontos foram resolvidos na Sprint, mas não foram planejados no Planejamento da Sprint;
- g) Mudança de escopo: calculada através da relação entre os pontos resolvidos e não planejados por pontos resolvidos;
- h) Meta máxima de mudança no escopo: uma meta para o máximo de mudança de escopo aceitável em uma Sprint.

Como observado com os dados apresentados na Tabela 1, a felicidade medida ao decorrer do projeto aumentou devido a gestão efetiva das pessoas, por mais empecilhos impostos ao time de desenvolvimento, como por exemplo a distância e a falta de encontros regulares na semana, os integrantes superaram e desenvolveram seguindo pilares importantes do Scrum, como é o caso da Transparência apresentada por cada um dos membros individualmente sobre suas opiniões, dificuldades e necessidades ao longo do desenvolvimento do projeto. Além da gestão efetiva do *team* e das análises métricas do projeto, a adaptação da metodologia com a inserção do D.D.S apresentou resultados positivos, visto que com o D.D.S, conseguimos obter produtividade com equipes de desenvolvimento dispersas. Unindo, D.D.S e o Scrum, o resultado fora positivamente alto,

dado produtividade, agilidade e possibilidades que métodos ágeis possibilitam. Possibilidades que dizem respeito a união e adaptação conforme a sua real necessidade, sendo principalmente a adaptação, um ponto chave utilizado no GPES. As conclusões obtidas após a inserção do Scrum no projeto serão relatadas no item abaixo.

5.3 Conclusões obtidas com o estudo de caso

O estudo de caso prático, analisado e vivenciado no grupo, demonstra não somente uma simples utilização da metodologia ágil de desenvolvimento de *software*, mas demonstra e comprova os benefícios da união entre o Scrum, a gestão efetiva de pessoas e a união do D.D.S. A principal dificuldade de acompanhamento do Scrum para o grupo foi justamente a distância e a falta de mais reuniões presenciais, porém com a utilização de técnicas e análises, as dificuldades foram sendo removidas. Os resultados apresentados aqui, dizem respeito ao alcance benéfico que a compreensão da gestão e da análise de dados referentes ao time oferece ao projeto, analisar métricas se torna fundamental para a avaliação do projeto. Nesse aspecto, o *Scrum Master* possui um trabalho de adequação e inserção de uma cultura ágil ao seu time de desenvolvimento, dependendo exclusivamente da importância com que o mesmo “enxerga” seus membros e o valor que ele estabelece aos mesmos, são pontos chaves que irão fazer com que os resultados da implantação sejam positivos.

6 CONCLUSÃO

Analisando as metodologias ágeis, se nota que elas pouco têm de novo, porém, quando se diz respeito ao seu enfoque, consegue-se notar que ele muda totalmente, de processos para as pessoas. As metodologias ágeis não mudam somente o enfoque, mas também mudam a maneira de como se passa a enxergar os processos envolvidos em projetos. A adoção de uma cultura ágil em projetos influencia de uma maneira direta o desenvolvimento do mesmo, observa-se não somente uma rapidez em termos de implementação, mas também pode-se analisar os resultados positivos apresentados pelo time de desenvolvimento, uma melhoria na relação time e gerência, pois com a cultura ágil, cria-se um time auto gerenciável onde as pessoas que estão presentes no processo de

desenvolvimento passam a se gerenciar por si próprias, retirando a necessidade de um gerente de projeto por de trás do mesmo. Devido a fatores como o alto índice de produtividade a auto gerência e a agilidade em termos de desenvolvimento, as metodologias ágeis estão cada vez mais sendo aceitas e inseridas ao meio de desenvolvimento de produtos de *software*.

REFERÊNCIAS

Gil, Antonio Carlos. **Método e técnicas de pesquisa social**. 6. ed. São Paulo: Atlas S.A, 2008.

Layzell, P.; Brereton, O. P.; French, A. **Supporting collaboration in distributed software engineering teams**. In: Asia-Pacific Software Engineering Conference, 7., Singapore, 2000. Anais Eletrônicos. Disponível em: <<http://www.computer.org/>>. Acesso em: 23 abr. 2018.

Pressman, Roger S. **Engenharia de software: uma abordagem profissional**. 7.ed. – Porto Alegre: AMGH, 2011.

Sbrocco, J. H. T. C.; Macedo, P. C. **Metodologias ágeis: engenharia de software sob medida**. 1. ed. São Paulo: Érica 2012.

SCHWABER, Ken; SUTHERLAND, Jeff. **The definitive guide to scrum: the rules of the game**. 2013. Disponível em: <<http://www.scrumguides.org/docs/scrumguide/v1/scrum-guideus.pdf>>. Acesso em: 09 abr. 2018.

Siqueira, Fábio Levy. **O desenvolvimento distribuído de software: características e recomendações para a gerência de projetos**/F.L.Siqueira. -- São Paulo, 2005. 152p.; Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Computação e Sistemas Digitais.

Sommerville, Ian. **Engenharia de Software**. 9. ed. São Paulo: Pearson Education do Brasil, 2011.