

A FUNÇÃO DO SCRUM NA GESTÃO DE PROJETOS DE SOFTWARE

EDSON COUTINHO DA SILVA

CENTRO UNIVERSITÁRIO DA FUNDAÇÃO EDUCACIONAL INACIANA PE SABÓIA DE MEDEIROS (FEI)

LUIZ RICARDO DE LIMA

CENTRO UNIVERSITÁRIO DA FUNDAÇÃO EDUCACIONAL INACIANA PE SABÓIA DE MEDEIROS (FEI)

A FUNÇÃO DO SCRUM NA GESTÃO DE PROJETOS DE SOFTWARE

1. Introdução

O conceito do *Scrum* surgiu por meio de Hirotaka Takeuchi e Ikujiro Nonaka em 1986, no qual publicaram um artigo na revista *Harvard Business Review*. Em virtude de suas experiências na área automobilística e tecnológica, eles introduziam um modelo que aumentava o desempenho no desenvolvimento de novos produtos comerciais, abordaram também, uma semelhança entre o modelo criado com uma jogada do esporte *rugby* (Takeuchi, Hirotaka & Nonaka, 1986; Sabbagh, 2013; Silva & Lovato, 2016).

O *Scrum* aparece em um segundo momento com os autores DeGrace e Stahl, no qual em 1990 os autores criticavam o modelo *waterfall*, mais conhecido como cascata, uma vez que enxergavam o modelo apresentado por Hirotaka e Ikujiro como possível solução para a área de desenvolvimento e de gerenciamento de projetos. Entretanto, o método somente foi consolidado por meio de Schwaber e Sutherland e somente em 1995 o *Scrum* é apresentado como um método de desenvolvimento de *software* ágil baseado no sistema *Lean* da Toyota (Machado & Medina, 2009; Silva & Lovato, 2016).

Assim, o *Scrum* surge com o propósito de suprir as necessidades que os profissionais da área encontram ao utilizar o tradicional método preditivo do *waterfall*, pois, na maioria das vezes, aquilo que é solicitado pelo cliente, ao término do projeto, não era o produto esperado pelo cliente, gerando um enorme custo e transtorno para ambas as partes. Constata-se que o *Scrum* se difere, por atuar em ciclos, conhecido por *sprints*, que normalmente possui duração de quatro semanas, e ao término da *sprint*, o cliente poderá utilizar ou não aquilo que foi entregue, em menor tempo, sem gerar grandes expectativas em relação à entrega.

Atualmente, este *framework* está se tornando mais comum no mercado, dado ao momento em que as empresas se encontram em constantes mudanças, fazendo com que elas, em uma enorme velocidade se adéque as necessidades de seus clientes. Assim sendo, quanto mais rápida for a resposta das empresas, maior será a satisfação em resposta ao mercado. Nesse contexto, o *framework Scrum* parece ser altamente indicado, se comparado aos métodos tradicionais.

Diante disso, este pesquisador percebeu a necessidade de um estudo empírico, e para tanto, o seguinte problema de pesquisa foi concebido: De que maneira os colaboradores de uma consultoria de tecnologia da informação (T.I.), localizada na cidade de São Paulo (SP), desenvolvem soluções em softwares para os seus clientes mediante *framework Scrum*? Com vistas a buscar respostas para este problema, o seguinte objetivo foi proposto: Compreender a maneira como uma consultoria de T.I., localizada na cidade de São Paulo aplica o *framework Scrum* para desenvolver soluções em softwares para seus clientes. Neste estudo, a perspectiva dos colaboradores foi considerada segundo métodos qualitativos.

No transcorrer deste trabalho, será apresentado um levantamento teórico acerca do *Scrum*, bem como os desafios enfrentados por empresas, o perfil esperado do time, a metodologia empregada para a organização do estudo empírico, os resultados obtidos através das entrevistas com profissionais da empresa, relacionando com o fator da cultura organizacional, a utilização do *framework*, a relação entre os *stakeholders* envolvidos, a contribuição e fatores para a geração de inovação, aspectos quanto os prazos de entrega, geração de valor, a relação com o Retorno do Investimento (ROI) e por fim, as considerações finais do estudo.

2. Referencial Conceitual

2.1 O Framework Scrum

Há de se reconhecer que a nova abordagem no cenário empresarial e a alta rotatividade de inovações tecnológicas, o uso de *frameworks* e metodologias ágeis estão sendo fundamentais para a sobrevivência das empresas de todo o porte e tamanho, no âmbito nacional e internacional. Com efeito, o interesse por tais ferramentas de gerenciamento de projeto tem elevado o ritmo no qual o mercado se comportou no passado, obtendo-se a partir de então, melhorias significativas e acirrando ainda mais o mercado de desenvolvimento de software (Duane, Richard & Heinz, 1999; Boehm, 2006).

Os fundadores do *Scrum*, Schwaber e Sutherland (2017), afirmam que, o *Scrum* se estabelece e se incorpora em um conjunto de conceitos e práticas já abordados em projetos, contudo, necessitam de flexibilidade, adaptabilidade e agilidade tornando o *Scrum*, mediante essas características, um *framework*. Ademais, o *Scrum* tem se tornado de grande valia no desenvolvimento de *software*, produtos, serviços e, também, no gerenciamento de empresas, resultando em constante valor entregue em uma pequena parcela de tempo, contento um alto padrão de qualidade (Schwaber & Sutherland, 2017). Mas, o grande marco ocorreu em 2001 em Utah (Estados Unidos), quando o *Agile Manifest*, foi concebido por um grupo de desenvolvedores e acadêmicos. O objetivo era descrever uma forma de transformar o desenvolvimento de *software*, com foco na rapidez e eficiência. Assim, desde então o *Scrum* tem revolucionado o mercado de desenvolvimento de *software* (Dingsoyr et al 2012).

Com efeito, o manifesto ágil se tornou um marco para o desenvolvimento de software sobre a mudança cultural das organizações e de quebras de paradigmas existentes no desenvolvimento de projetos. O escritor Brod (2013) reforça em seus estudos que o manifesto ágil deve ser incorporado por toda a equipe de forma que seja um comportamento natural por todos. Já Vieira (2014) descreve metaforicamente, que o *Scrum* representa “*a fundação e as paredes de um edifício*”, de forma que, não devem ser alterados sem correr o risco de um eventual colapso, sendo possível o incremento de suas interações de acordo com cada organização, desde que, não sejam alteradas suas estruturas principais. (Beck et al, 2001; Machado & Medina, 2009).

Assim, o *Scrum* é um *framework* ágil que pode ser aplicado ao gerenciamento de projetos de desenvolvimento de *software*, serviços e, também, no gerenciamento da própria empresa, inseridos ou não em ambientes complexos. O *Scrum*, é embasado no empirismo, com uma abordagem interativa e incremental, a fim de entregar valor com frequência, diminuindo os possíveis riscos no decorrer do projeto, com apenas três pilares para sua implementação de controle empírico: transparência, inspeção e adaptação. (Cooper, 2014; Sabbagh, 2013; Schwaber & Sutherland, 2017)

2.2 Scrum na Prática

O início do projeto se dá a partir da visão do produto, no qual o *Product Owner* elabora uma lista ordenada, incompleta e dinâmica com a forma esperada que o produto se desenvolva ao longo do projeto, esta lista é chamada de *product backlog*. Os itens do alto no *product backlog*, possuem um número maior de requisitos a serem desenvolvidos e trabalhados, sendo de maior importância para o projeto, enquanto os itens mais baixos, uma quantidade menor de detalhes. A lista poderá sofrer alterações em todo o projeto, podendo surgir novos requisitos a qualquer momento. (Sabbagh, 2013; Silva & Lovato, 2016).

Em seguida a criação do *product backlog*, ocorre uma reunião chamada *sprint planning*, nela participam às três partes envolvidas (ou *stakeholders*) e será definido o que será feito, e o que o *Scrum Team* aceitou durante o planejamento. Após aceito, o mesmo não poderá ser alterado até o termino da *sprint*. Uma característica típica do *Scrum* é o conceito de tempo fechado (*timebox*) e as *sprints*, que possuem tempo máximo de conclusão de até quatro semanas ou trinta dias. Cada *Sprint* possui uma parte do produto final, que resulta ao cliente a possibilidade de testar, utilizar e receber parte do seu investimento rapidamente. Isso proporciona ao cliente valor, diferentemente de outras formas de gerenciamento de projeto, como, por exemplo o PMBoK (*Project Management Body of Knowledge*)

que somente após a entrega do projeto o cliente poderá testar ou utilizar o produto e em muitos casos o que foi entregue não era o esperado (Carvalho & Mello, 2012; Sabbagh, 2013).

Cabe apontar que uma *sprint* termina com uma *sprint review meeting*, uma reunião que as funcionalidades solicitadas foram implementadas e será demonstrada ao cliente. Após a entrega, acontece a *sprint retrospective*, onde são destacados ao time os pontos positivos e os pontos que ainda precisam ser aprimorados. Ao término de uma *sprint*, o *Scrum Team*, deverá ter gerado um produto segundo os itens do *sprint backlog* (o incremento do produto). Nesse momento, nenhuma modificação no produto é necessária para que o produto seja entregue, sendo que, cabe ao *Product Owner* aceitar a entrega no momento ou, ao término do projeto. Entretanto, se no decorrer de uma *sprint* o *Product Owner* perceber que a meta estipulada não será atingida e trará algum impacto, o mesmo poderá cancelar a *sprint* antes mesmo do prazo acabar, para dar início a próxima *Sprint*, embora, isso não seja esperado, pode vir a ocorrer (Machado & Medina, 2009; Sabbagh, 2013).

Convém esclarecer que sempre que o produto for entregue e, ainda, representar valor suficiente de modo que possa ser utilizado, é importante que seja disponibilizado rapidamente ao cliente para que o produto possa ser utilizado pelo usuário e logo retorne em valor para o cliente. Em geral o *Scrum* possui quatro artefatos, tais como: *product backlog*, *sprint backlog*, *burndown chart* e *impediment backlog*. O artefato *burndown chart* é um gráfico que apresenta a quantidade de trabalho restante, dia após dia. Neste gráfico quanto mais horizontal for a sua tendência, melhor poderá ser o resultado no desenvolvimento, pois a altura define a quantidade de *sprint backlog* não concluídas, já o artefato *impediment backlog* são os impedimentos que impedem o *Scrum Team* progredir. Esta lista pode ser classificada por: *Team impediment*, que são os impedimentos que o time pode resolver e também, o *Organization impediment*, são os impedimentos que estão fora do alcance do *Scrum Team*. Assim, cabe ao *Scrum Master* auxiliar e solucionar todos os impedimentos. (Machado & Medina, 2009; Silva & Lovato, 2016).

Todo projeto possui prazo de início e de término (*timebox*), entretanto, um projeto pode ser encerrado por diversas razões, tais como: encerramento por tempo de contrato, o cliente já está satisfeito com o conteúdo entregue (caso esteja estipulado no contrato), falta de pagamento, entre diversos outros motivos. Após a conclusão do projeto, é de se esperar que as partes mais importantes tenham sido entregues, logo, o projeto será bem-sucedido somente se a visão do produto for alcançada. (Sabbagh, 2013).

2.3 Desafios do *Scrum* em uma Empresa

Devido à cultura organizacional de algumas empresas possuírem formas mecanicistas de trabalho, a implantação do *framework Scrum* se tornaria difícil, devido as suas características serem orgânicas na sua forma de atuação. Assim, os profissionais precisam estar preparados, pois, o *Scrum*, é voltado para o *Scrum team*, que trabalham no desenvolvimento do *software* e não apenas para os processos. Desde modo, o *Scrum* se difere das demais metodologias tradicionais (burocráticas), que possuem uma enorme quantidade de documentação e que os requisitos, permanecem sem alterações. (Machado & Medina, 2009).

Um dos grandes desafios do *Scrum*, é com relação a mudança de *mindset* nas organizações e também, um assunto de bastante resistência no gerenciamento de projetos, uma vez que o mercado está altamente dinâmico e inovador, e cada vez mais, está terceirando a sua mão de obra em todo o planeta, indicando que, tais mudanças organizacionais e gerenciais, devem se voltar as ações e aspectos ágeis. (Denning, 2013; Silva & Lovato, 2016).

Date, Pinochet, Bueno e Nemoto (2016), discorrem sobre as mudanças na cultura organizacional após a implementação de tais métodos de gerenciamento, no qual, é notável a melhoria da organização em relação aos papéis atribuídos aos processos, na comunicação e na redução dos atrasos. Em contrapartida, Amaral e Conforto (2014) discordam dos mecanismos do *Scrum*, no que se refere, a gerenciar um número maior de projetos e equipes, no qual, afirmam não serem eficazes, devido, não poderem dimensionar e compartilhá-los com os demais projetos.

Já de acordo com Machado e Medina (2009) o *Scrum* não é milagreiro e muito menos irá oferecer uma receita pronta, para resolver todos os problemas no projeto. A única certeza é que esses problemas serão identificados com mais facilidade. Por se tratar de um *framework* servirá como um guia de boas práticas para alcançar seu objetivo. (Moe, 2012).

Logo, para que haja uma transição de modelos clássicos e tradicionais, para os modelos ágeis, é necessário que haja uma compreensão maior sobre a necessidade de cada organização, área e projeto, visando analisar o que será necessário implementar, para que não haja um fenômeno indesejado após a sua implantação. (Heitor, 2017)

2.4. Perfil do Time

Em tese, o *framework Scrum*, se consiste em três elementos principais: *Product owner*, *Scrum master* e *Scrum team*. Os requisitos do produto, são listados no *product backlog* pelo *product owner*, checados em uma reunião, chamada de *sprint planning*, no qual é realizada uma nova listagem, chamada *sprint backlog*, com os requisitos que serão realizados em cada interação (*sprints*). (Bassi & Filho, 2008; Schwaber & Sutherland, 2017)

Diariamente, ocorrem reuniões rápidas (*daily meeting*) para acompanhamento do processo. Após a conclusão da *sprint*, é realizada uma reunião chamada *sprint review meeting*, onde serão testados os requisitos pelo cliente; após a entrega da *Sprint*, ocorre uma reunião chamada de *sprint retrospective*, que serão analisados os pontos positivos e negativos ocorridos dentro da *sprint* e para o acompanhamento gráfico do projeto e de suas fases, é utilizado o *burndown chart*. (Silva & Lovato, 2016)

O *product owner* representa a figura do cliente, nele consiste em elaborar e priorizar os requisitos do sistema a serem desenvolvidos chamado de *product backlog*. Ele é o responsável pela maximização do ROI (*return on investment*) de acordo com a forma que prioriza os itens que possuem a maior relação favorável de valor agregado com os seus custos. A grande parte da complexidade externa, em relação ao desenvolvimento do produto, é centralizada no *product owner*, responsável pelo contato com os *stakeholders* e visa agilizar os processos. (Deemer, Benefield, Larman & Vodde, 2010; Sabbagh, 2013; Schwaber & Sutherland, 2017).

Já o *Scrum master*, possui a responsabilidade e autoridade direta de manter o funcionamento do *Scrum* no projeto, de forma que as suas práticas sejam implementadas e seguidas no decorrer do desenvolvimento, nele está a responsabilidade de orientar o *Scrum team* de forma que garanta a entrega de valor ao cliente e evitar que possíveis eventos externos obstrua ou impeça a realização das *sprints*. (Carvalho & Mello, 2012; Schwaber & Sutherland, 2017).

Assim, o *Scrum master* não assume a função de gerente de projeto, pois o mesmo não delega funções ou designa tarefas; no *Scrum*, cabe ao time de desenvolvimento determinar as distribuições das tarefas, de acordo com as habilidades de cada integrante. Desta forma o *Scrum master* é considerado como um facilitador ou orientador, o que torna o *Scrum team* auto gerenciável e multifuncional. (Sabbagh, 2013; Schwaber & Sutherland, 2017)

O *Scrum team*, é uma equipe composta por profissionais que atuam na área da programação, *designer*, teste, qualidade dentre outras áreas que são responsáveis em transformar os itens do *product backlog* em *sprints*. O time é composto entre 5 a 9 integrantes, devido à complexidade de relacionamento entre a equipe e do acompanhamento do *Scrum master*. O *Scrum master*, não determina o que será feito no momento da reunião, deixando com que os integrantes do time se interajam e juntamente cheguem a conclusão do que será feito, mas, caso ocorra algum problema entre o *Scrum team*, o *Scrum master* poderá intervir. (Sabbagh, 2013; Schwaber & Sutherland, 2017)

Outro detalhe importante, são as reuniões diárias chamadas de *daily Scrum*, com duração máxima de 15 minutos, que são realizadas em pé (*stand up meeting*) logo pela manhã, e sempre no mesmo horário, no qual o *Scrum master*, juntamente com o *Scrum team*, se baseiam em três perguntas centrais, tais como: O que foi feito no dia anterior? O que será feito nas próximas 24 horas? Ou se existe algum impedimento que poderá interromper o processo de desenvolvimento? (Pressman, 2006; Sabbagh, 2013)

3. Metodologia

Este estudo é classificado como descritivo, pois, procura apresentar uma ampla descrição da forma empregada e de pontos de vistas diferentes no uso empregado do *framework Scrum* em uma empresa multinacional de consultoria de TI (tecnologia da informação) localizada na cidade de São Paulo, no qual, desenvolvem soluções e *softwares* para seus clientes. Este estudo é delineado como estudo de caso, pois, foram realizadas algumas pesquisas de campo, de maneira que, este pesquisador, diante do tamanho da empresa, teve acesso restrito e somente de leitura em alguns documentos, apresentações e entrevistas. Dentre os funcionários, foram selecionados 7 entrevistados, dada a importância destes profissionais para a organização, pois, foram os percursores na implantação deste *framework* e também, para o funcionamento do *Scrum* em um projeto no qual este pesquisador atua. (Minayo, 2001; Gil, 2009).

No tocante ao campo de estudo, esta organização possui no Brasil, cerca de 1.800 funcionários, além dos *off-shores* e outras unidades em outros países, que também atuam em projetos. Aproximadamente 70% atuam nas diversas áreas técnicas enquanto os demais em áreas de gestão, negócio e administrativas. A empresa é subdividida em verticais e horizontais; as verticais, possuem áreas diferentes de atuações e possuem variados projetos com diversos segmentos. Já nas horizontais, prestam apoio e suporte para as verticais, como, por exemplo, o setor financeiro, recursos humanos, infraestrutura, dentre outros.

Utilizando os três elementos fundamentais para execução do *Scrum* em um projeto, de acordo com a literatura, os entrevistados possuem tais funções: 1 *Scrum master* (SM), 1 *product owner* (PO), os integrantes do *Scrum team*, sendo 1 desenvolvedor (DEV) e 1 programador (PA), 1 *product leader* (PL) e com uma visão tradicional, com algumas práticas do *Scrum* 1 *project manager* (PM). Ambos foram selecionados, pois representam as principais figuras dentro do *Scrum* e também, com o propósito de compreender os processos de cada unidade dentro do *Scrum*, de forma prática e interativa, abordando os três elementos (chave) sendo o *product owner*, *Scrum master* e *Scrum team*.

Também, foram convidados para a entrevista 1 *agile coach leader* (ACL), 1 *engagement leader* (EL) e 1 *project manager* (PM), pois, além dos elementos principais para o funcionamento do *Scrum*; este estudo visa também, compreender e descrever os benefícios e dificuldades de gerenciar uma equipe utilizando o *Scrum* e como é o contato com os clientes apresentando este *framework*.

Sendo um estudo qualitativo, foi elaborado um roteiro de entrevista com base em 6 tipos de abordagem centrais: (1) Estrutura *Scrum*, (2) Cultura organizacional, (3) *Stakeholders*, (4) Inovação, (5) Prazo de entrega e (6) ROI. Tais abordagens, serão utilizadas para análise e compreensão dos resultados obtidos através da percepção deste pesquisador no campo de pesquisa por intermédio das entrevistas e também por meio das transcrições obtidas através da fala dos entrevistados.

Os dados coletados, foram obtidos por meio de entrevistas, que ocorreram em 4 períodos diferentes, entre os meses de maio e outubro de 2018. As entrevistas foram conduzidas, por meio da ferramenta Cisco-Webex (plataforma de comunicação remota), devido alguns participantes estarem alocados em outros clientes, não sendo viável naquele momento fazer as entrevistas presenciais, logo, está opção foi a mais coerente para aplicação do questionário com 12 questões, e foram aplicadas primeiramente ao *Engagement Leader* (EL); no segundo período, foram aplicadas às seis primeiras questões ao *Scrum team*, composto por um *Scrum master* (SM), um *product owner* (PO), um *developer* (DEV), um *programmer Analyst* (PA) e um *project manager* que aplica o *Scrum* no

gerenciamento do time e de suas atividades (PM); no terceiro período, foi aplicado o questionário para um *Agile Coach Leader* (ACL).

No quarto período, foram aplicadas às seis últimas questões que estavam pendentes do segundo período ao *Scrum team*, entretanto, devido às dificuldades de agenda, férias, entre outros motivos, alguns participantes não estiveram presentes e outros foram incluídos, para não perder o rumo deste trabalho, sendo este último período composto por um *product leader* (PL), um *developer* (DEV) e um *programmer analyst* (PA).

4. Análise dos Resultados

4.1 - Estrutura *Scrum*

De acordo com o ACL, as pessoas precisam saber aquilo que precisa ser feito, pois, o *Scrum guide* não define as posições e todos devem atuar juntos, não havendo separações, muito menos a figura do líder na estrutura. Em sua visão de mercado, o mesmo afirma ser interessante a figura do *developer* e do *tester*, de forma que ambos se ajudem mutuamente, sem separações de funções de forma egoísta, e essa sensibilidade, deve partir por parte do time, e não algo imposto a ser feito, o *Scrum master* pode ajudar, mas cabe ao time agir desta forma.

Segundo o EL, não é possível replicar o *Scrum team* como se fosse algo simples, de forma que, acaba não sendo possível pegar um time que funciona bem e montar outro que funcione igual. O EL enfatiza que, isso ocorre pelo fato de que cada time possui pessoas, e cada pessoa é diferente uma das outras, partindo desse ponto, elas se complementam como um só time. Para ele, o segredo em montar bons *Scrum teams*, está de acordo com o que foi abordado pelo ACL, que não se dá pela divisão de tarefas, pois, segundo o *Scrum guide*, não existe este tipo de blindagem, e sim que o time esteja aberto, por ser de fato um *framework* e não um método.

De uma forma geral, ele acredita que a divisão de papéis seja algo importante, mas, acima disso, está a importância dos perfis nas pessoas, com a ênfase no perfil comportamental, que, em sua opinião, o fator principal da estrutura *Scrum*, é que os perfis se complementem, e isso se dá, pelo fato de todos dentro do time, estarem em busca de uma meta para o projeto e não mais de uma meta pessoal. E quando ocorre de ter dentro do time, alguma pessoa egocêntrica, não é possível ter um bom time, pois, a pessoa em si, não está preocupada com os membros ou com o projeto, ela está preocupada consigo mesma, com as contas que tem a pagar, e esse tipo de perfil, precisa muito ser manejado.

Já para o PO, ele sugere que se deve seguir os princípios do *Scrum guide* e ter o *product owner* com o poder de decisão, ter o *Scrum master* como um facilitador, sendo a figura responsável por tirar os impedimentos, e ajudar a todos no projeto; ter um *Scrum team* com os *skills* necessários para o desenvolvimento da *sprint*, mas o que acontece realmente, é a atuação do *Scrum master* em vários projetos ao mesmo tempo, e o *product owner* muitas vezes, não tem o poder de ser o intermediário entre os *stakeholders* e o time.

Na visão do SM, seria necessário ter uma estrutura envolvendo todas as tecnologias necessárias, e segue de acordo com o que foi abordado pelo PO, que em muitas vezes, não é o que acaba acontecendo, já, em contrapartida, o PA acredita que, a forma como o *Scrum team* deve ser estruturado, se dá pelo fato de ser auto gerenciado, em que, o time fica mais livre para poder atuar e aborda a dificuldade de pessoas que não são tão bem auto gerenciadas e que estas pessoas, precisam ter um pouco mais de prática.

Segundo o DEV, o time deve ser relativamente pequeno, multidisciplinar, assim como aborda o *Scrum guide*, e isso seria o mundo **ideal**. Após a formação de uma equipe, ela deve ser igual a um time de futebol, eles precisam aprender a trabalhar em campo juntos, e a analogia com o esporte, no *Scrum* é inevitável, pois, a base do *Scrum* é provinda do esporte.

Já para o PM, o time precisa ter a liberdade para opinar sobre as soluções; atuar com um bom desenvolvedor, e às vezes, não é preciso ter um especialista alocado, mas alguém que tenha uma boa visão técnica, a fim de cumprirem bem os seus papéis dentro do projeto.

Com base na forma que é feito o processo de avaliação dos processos do *Scrum*, o ACL utiliza o MPS (*Master Production Schedule* ou Programa Mestre de Produção) no qual é medido o nível de satisfação do cliente e a satisfação da equipe. Ele utiliza também o *Burndaw* de *task* e o *Chart* de histórias que contém o *backlog*, ele utiliza também o Moscou e divide as histórias em *Moust*, o que precisa ser feito; *Could*, o que pode ser feito; *Should*, o que poderia ser feito, e *Won't*, o que não vai ser feito. Para ele, a primeira coisa que se deve ser avaliada, é o que está sendo entregue, se está entregando valor, se o que foi entregue era o esperado, se os objetivos estão sendo cumpridos, e isso também é um indicador, é a satisfação dos clientes. Ele acrescenta que os indicadores são objetivos que foram mapeados e estão dentro do *backlog*, se os programas que estão saindo, qual seria o nível e a quantidade de *bugs* que estão saindo após-prod, e isso é a satisfação do cliente; com relação ao time, a avaliação é realizada o tempo todo, nas reuniões de retrospectiva, e também, é verificado o nível de satisfação do time e como eles estão.

Para o EL, ele chegou a utilizar métricas de *compliance* em uma planilha do Excel, e havia como indicador, uma bolinha vermelha para triste, uma bolinha verde de feliz e uma bolinha amarela de mais ou menos para vários requisitos que o time acreditava serem importantes, para o *compliance* da célula *Scrum*. Isso era aplicado em várias etapas do processo, e segundo o EL, essa forma de avaliação não é uma forma indicada, pois, acabava gerando uma expectativa e conseqüentemente, oprimia a pessoa pelo fato de estar medindo. Segundo o EL, a proposta do *Scrum* é deixar a pessoa com liberdade, ao invés de ficar cercado e monitorando.

Ele acredita que a relação da meritocracia e bonificação, deve ser bem avaliada quando se opta por utilizar o *Scrum*, porque o bônus pode se tornar muito mais ameaçador, do que algo que possa facilitar o processo, por isso acaba sendo algo complicado. Ele menciona também a utilização da ferramenta *Ready*, utilizada desde a construção da história, até a estrutura do próprio *Scrum* no projeto, e relata da importância do *coaching* no processo de estruturar o *Scrum team* de forma madura, devido os profissionais serem muito mais técnicos do que humanos, e normalmente, quem realiza estes processos de avaliação são os *coachings* e o *Scrum master*.

De acordo com o PL, a ferramenta principal seria o quadro Kanban, que deve mostrar visualmente, para qualquer pessoa (leiga ou não) em que período e momento que o projeto está, como se fosse passar algo para qualquer uma pessoa. Segundo o DEV, ele está de acordo com o PL e aborda a ferramenta *planning poker* para entender a estimativa, e após a aplicação do *planning poker*, buscar entender em que ponto a análise inicial falhou, em que ponto time errou a estimativa e após isso, realizar o ajuste para as próximas histórias, buscando chegar ao nível mais próximo do real.

Acrescentando o que foi abordado pelo ACL, o EL apresentou o modelo *Cynefin*, criado com base na análise da complexidade de sistemas (sistemas humanos) que define algumas métricas de decisões para a utilização de métodos ou *frameworks* em diferentes cenários, e isso facilita bastante no momento de decidir com o cliente, qual seria a melhor estratégia e forma de prosseguir com o projeto, se é utilizando os *frameworks*, projetos tradicionais ou com um *mix* deles, e para gestão a vista, ele sugere a utilização do Kanban.

4.2 Cultura Organizacional

Em relação à cultura organizacional da empresa, com o uso do *Scrum*, segundo o ACL, as empresas que possuem um alto sistema de hierarquia e cultiva a figura do chefe, piora o real uso do *Scrum*, e quando existe a cultura do chefe, existe a figura do aponta dedo. Ele afirma que as pessoas não se contentam em ver o problema, elas querem achar o culpado, e quando ocorre esse processo, quebra-se um dos princípios do *Scrum*, que é a auto-organização da equipe e também sobre o terceiro princípio ágil, que é a colaboração com o cliente, e conta com a negociação de contratos, ou seja, elas precisam estar juntas, por isso, quanto mais hierarquizada a empresa for, piora este processo e passa a existir mais dificuldades, pois, não existe confiança.

Ele acrescenta que, quando se fala de comando e controle, parte do pressuposto a uma camada de pessoas ignorantes, motivo pelo qual são aplicados o comando e o controle, relacionados a falta de

confiança, pois, não existe delegação ou gerenciamento, e sim, o que de fato aquela pessoa está fazendo ou produzindo, e isso é a falta de confiança. Segundo o ACL, este tipo de comportamento é muito típico no Brasil, o fator da desconfiança, em achar que o desenvolvedor não está trabalhando quando se fala em manter o escopo aberto no projeto, onde o gerente de projeto não aceita o *Scrum* de início, por causa da desconfiança, sendo que no *Scrum*, ele irá acompanhar todo o processo, não havendo motivos concretos para tal comportamento.

Outro ponto destacado, é quando a pessoa de negócios não quer se envolver no projeto e acaba complicando muito o desenvolvimento e andamento dos processos, pois, acredita que a ideia que ele teve, que foi passada para o papel, vai chegar do mesmo jeito para o programador, e este programador não entendeu muito bem o que foi solicitado, vai tentar adivinhar e fazer aquilo que entendeu, por causa que a parte mais interessada, não quer se envolver no projeto.

Já em relação às facilidades, ele descreve que, a facilidade ocorre, quando a empresa já está preparada para trabalhar com a gestão à vista; em um projeto no qual ele estava atuando, a empresa já estava com a cultura *Lean* muito forte, e isso ajudou bastante na utilização do *Scrum*, e quando se tem o *mindset Lean*, facilita este processo. Outro ponto citado por ele, que parece um paradoxo, é quando a empresa não tem muito dinheiro, e isso facilita o trabalho com o *Scrum*, pois, quando não se tem muito dinheiro, a empresa precisa entregar alguma coisa de forma bastante rápida no mercado, ela não pode se dar ao luxo de esperar um ano para ficar pronto o projeto, então, ela precisa de algo bastante ágil, e o *Scrum* é a salvação para este problema.

Já em relação as grandes empresas, ocorre uma concorrência de egos do que em relação a parte financeira, eles preferem ficar dois anos com um projeto, sem saber se no final, irá funcionar ou não, querem economizar ao não ceder um funcionário para ajudar no processo, mas preferem arriscar todo um projeto de dois anos para ter o escopo fechado.

O ponto de vista do EL, também tem relação com o que foi apresentado pelo ACL, sobre a cultura brasileira, no qual a cultura organizacional, acaba sendo um pouco do reflexo da cultura do país. Ele apresentou um ponto que seria bom e outro ruim, sendo o ponto ruim, uma cultura pobre de valores por conta dos maus hábitos, como cortar uma fila, passar na frente dos outros, tendo como um ponto principal, a motivação do trabalho para pagar contas. Ele enfatiza que esta postura é um câncer dentro das organizações, dentro de uma cultura, dentro do Brasil ou de qualquer país que queira se desenvolver, e que as pessoas, não deveriam trabalhar somente com esta intenção.

Acrescenta também, que pessoas que trabalham com esta visão, são pessoas desmotivadas intrinsecamente, que só estão atrás do dinheiro, não entendem que o trabalho que exercem podem transformar ou contribuir em relação as outras pessoas na sociedade, são pessoas com a chama apagada, sendo um problema bastante sério; a pessoa precisa saber o motivo de estar viva, encontrar o seu propósito de vida, pois sem isso, o trabalho jamais fará sentido algum. Já em relação à cultura, o EL menciona, que é uma característica interessante, pois o brasileiro possui uma grande diversidade, flexibilidade, adaptabilidade e criatividade, e essa criatividade, aumenta a capacidade de inovação, sendo alguns dos pontos positivos da cultura brasileira.

De acordo com o PO, a empresa precisa estar disposta a mudar a forma com a qual trabalha, precisam estar abertas para inovar, e mesmo quando o cliente aceita a implantação, conhece o processo e inicia o *Scrum*, é necessário que haja paciência, pois, acaba sendo normal que a primeira *sprint* não saia como planejado, mas conforme as demais *sprints* vão ocorrendo, se torna muito melhor a qualidade, na medida que time vai se desenvolvendo, e nas *sprints* seguintes, possa funcionar melhor, mas não é algo imediato.

Na opinião do SM, a grande dificuldade do cliente é de planejar a demanda, descreve que nem sempre possuem um *backlog* detalhado, investigado e que consigam repassar estas informações de forma concreta, informa sobre a dificuldade de conseguir alcançar este ponto, sem que haja um bom planejamento e também, de saber pedir o que realmente se espera. Já em relação à facilidade, seria em relação à quantidade de recursos para se iniciar o processo de desenvolvimento e evolução do projeto, e isso, facilita no processo de futuras alterações, devido a sua documentação ser mais enxuta.

De acordo com o DEV e o PM, ambos concordam com a visão do PO no qual se deparam com uma grande dificuldade quando, o cliente não está por perto do time para poder esclarecer eventuais dúvidas, que em determinados momentos, são cruciais para a evolução do projeto. E quando não há a participação do cliente, o time se sente completamente à deriva em relação a estas informações que podem fazer toda a diferença na entrega, além da facilidade de poder fazer as entregas segmentadas, com *deliverys* funcionais segmentados, além da documentação mais enxuta, acrescenta o PM.

Com relação à quebra ou não dos paradigmas, o ACL explica que um dos paradigmas é a empatia; empatia em relação a equipe de negócios em ajudar a área de IT, no sentido que, IT passa a se responsabilizar por alguns pontos técnicos e não com etapas, em relação aos processos do projeto, se algo será entregue ou não, isso é de responsabilidade de negócios. Ele descreve sobre a importância dos testes serem realmente efetivos, e não quando um sistema está burlado, principalmente quando se passa de um projeto cascata e passa para a implantação do *Scrum*, pois, se no formato tradicional constar uma quantidade bastante inferior, com o *Scrum* os testes acontecem a todo o momento, e seria algo questionado pelo cliente, por isso a preocupação da veracidade dos processos.

Outro ponto que o ACL destaca, é a figura do aponta dedo, que também é um dos paradigmas a serem superados, quando a figura do líder é bastante constante na organização, ele acrescenta que não existe um líder exclusivo, todos são líderes, e acontece de forma orgânica, pois, cada pessoa possui um temperamento diferente em cada situação, por isso não se deve ser imposto.

Segundo o EL, um dos paradigmas bastante complicado é em relação ao escopo fechado, informando sobre a dificuldade encontrada no Brasil e América latina em se trabalhar com contratos com o escopo fechado, preço fechado, tudo fechado na utilização do *Scrum* que são processos vivos que estão sobre constantes processos de melhoria ao longo das *sprints*, pois aquilo que foi escrito hoje, amanhã o cliente e a equipe terá uma ideia muito melhor sobre aquela ideia inicial, por isso é um paradigma.

Um outro ponto bastante importante, é com relação ao gerente de projetos, acrescenta, questionando qual seria o papel dele, no projeto? Pois no *Scrum*, não existe esta figura e segundo o EL, o gerente de projeto acha ser um demérito se tornar um *Scrum master* por causa dos seus longos períodos de estudo, técnicas e certificações que, quando se depara com algumas dezenas de páginas do *Scrum guide*, não se dá muita importância, por possuir um perfil mais técnico do que humano, e em muitos casos, o gerente de projetos não tem o perfil de ser um *Scrum master*, pois, ele está acostumado a usar o chicote ao invés de deixar com que o time se auto organize, sendo um outro paradigma a ser quebrado.

4.3 Stakeholders

Partindo da forma como os clientes enxergam o *Scrum*, e um dos pontos de resistência apresentado pelo ACL, é o preconceito; ele descreve alguns casos no qual o modelo cascata predomina, por ser completamente racional e fechado, e isso transmite segurança para quem contrata, mas, na verdade, o cascata só é bonito no papel, e quando se inicia o projeto, é quando desencadeia uma série de problemas, em relação à permanência dos recursos, processos, documentação, desenvolvimento entre outros, e quando é apresentado a estrutura do *Scrum* para quem sempre utilizou o cascata, o cliente se assusta ao manter toda equipe unida, por manter o teste do início ao fim do projeto, por ter a figura da pessoa de negócio acompanhando o projeto, isso é completamente diferente do cascata, logo, o cliente não está disposto a conviver com a incerteza, se aquilo realmente dará certo ou não, no cascata, dá a entender que tudo dará certo, pois, está planejado, no *Scrum* não, ele atua em cenários incertos, mas atua de forma clara e transparente, com a realidade exposta tanto para o cliente, quanto para o time, e todos fazem parte do sucesso ou não do projeto.

De acordo com o ACL, o grande segredo é envolver a pessoa de negócio no projeto, e isso complementa o que foi mencionado anteriormente pelo PL, o ACL descreve que, no momento que a figura de negócio compreende o *Scrum* e passa a atuar juntamente com o time, muda completamente

a forma como a empresa passa a enxergar o *Scrum* e começam a inserir o *Scrum* nos demais projetos. Ele explica também que, no momento que IT contrata algum serviço, a área de negócio não quer saber o motivo de nada, e sim o porquê, tal produto ou serviço não foi entregue e não está gerando resultados. No momento que a área de negócio começa a ser envolvida, ela sai da bolha que estava e então, começa a agregar valor para a empresa e para o projeto como um todo, e isso só acontece quando a figura de negócio está envolvida, é a figura chave.

O EL menciona que hoje, os clientes acreditam mais no *Scrum* do que em alguns anos atrás, conforme foi mencionado pelo PA em relação à aceitação, entretanto, após o período de saudações, boas vindas e início das atividades, começam a surgir algumas perguntas sobre os integrantes, processos e sobre a pessoa de negócio no projeto, e um dos momentos que acaba deixando o cliente ainda mais intrigado, é com relação aos gerentes de projetos, que não faz a composição do *Scrum*, e segundo o EL, o gerente de projeto, é a figura que possui o chicote e a batuta, e sai controlando o tempo no manômetro, o que faz a cadência e sai batendo o bumbo, e o *Scrum* ajuda a quebrar essa estrutura, a começar com o gerente de projetos, pois, ele é essencial em outras áreas, como nas metodologias, mas não no *Scrum*. Eles até podem se tornar *Scrum máster*, e alguns até vão, pois, entendem que está é a liderança do futuro, e está muito mais em motivar as pessoas, do que, em ter que controlá-las.

Na perspectiva do PO, o cliente enxerga o *Scrum* como algo inovador, e que resolverá todos os seus problemas, e ele realmente pode resolver, mas segundo o PO, essa resolução não seria tão rápida em relação à expectativa do cliente, fazendo com que está expectativa seja um grande problema. Em relação ao SM, ele concorda com o que foi abordado pelo ACL e EL quanto a estrutura hierarquizada das empresas, e segue afirmando que, quando o cliente possui a sua estrutura e processos ultrapassados, o *Scrum* é muito bem-visto, que seria algo milagroso ou aquilo que ajudará a salvar os projetos, porém, ele não é mágico. Ele afirma também, que o cliente necessita estar preparado para começar a utilizar o *Scrum*, deve estar aberto para as eventuais mudanças, e comprar realmente a ideia, pois, caso contrário, o *Scrum* não dará certo; um ponto interessante apresentado pelo SM, é quando o cliente procura as empresas em busca do *Scrum*, e quando isso ocorre, aí sim! o projeto vai bem.

Quanto ao PM, o ponto que ele destaca é com relação ao custo, devido o cliente estar acostumado com os projetos tradicionais, que possuem o escopo fechado e não querem no decorrer do caminho realizar algum tipo de alteração, e com o uso do *Scrum*, acaba sendo inevitável as mudanças, devido em algum momento precisar alterar a quantidade de horas que estavam previamente estipuladas.

4.4 Inovação

Em relação à forma que o *Scrum*, proporciona ou agiliza o processo de inovação no desenvolvimento de software, segundo o ACL, o *Scrum* possui os mesmos princípios do *Lean*, no qual é necessário planejar, desenvolver, checar e ajustar. Segundo o ACL, inovação é algo novo, é algo que ninguém conhece e que pode dar muito certo, ou pode não funcionar.

Ele menciona também, sobre a incoerência em comparar inovação com sucesso, sendo que o correto seria inovação x tentativa, no qual o risco é elevado e acrescenta sobre a utilização do *Scrum* no processo de inovação, pois, do início ao fim do projeto, ocorrem testes constantemente, e logo que é identificado um problema, surge a oportunidade de acrescentar algo novo, contribuindo para a inovação. Entretanto, somente quem poderá dizer se aquilo que foi desenvolvido é inovador ou não, é o mercado, e isso não passa pela decisão de especialistas ou profissionais da área, somente o mercado tem esta autonomia.

O segundo ponto que ele aponta, é em relação ao cascata, no qual, a pessoa de negócio, só verá o resultado do projeto após estar com status concluído, ou seja, existe uma enorme probabilidade do cliente não ficar satisfeito com o que foi entregue, no *Scrum* este processo de avaliação da pessoa de negócio acontece a todo instante, aquilo que foi solicitado a ser feito, será realizado em *sprints* e

com isso, o cliente terá uma base sólida do que está recebendo e de como será o produto final; conseguir definir o que seria interessante entregar primeiro, é um enorme diferencial, maximizando a satisfação e resultados para o cliente, e por fim, ele aborda a questão de quanto mais cedo for testada a hipótese e o produto for lançado no mercado, maiores são as chances de se conseguir inovar e entregar algo realmente novo ao mercado, afirmando que no cascata, também é possível inovar, mas no contexto geral, o *Scrum* possui uma estrutura mais apropriada para o time inovar.

No ponto de vista do EL, ele apresenta três motivos de alguns times, serem mais inovadores se comparado aos outros; o primeiro motivo, se consiste no próprio *Scrum*, ele foi criado para fazer algo novo, comumente utilizado quando o escopo se encontra aberto; quando o projeto não possui um plano completo e o que predomina são expectativas, se existe mais ou menos a ideia do objetivo e não se sabe muito bem como fazer.

O EL vai de encontro com o que foi abordado pelo ACL, no sentido que o *Scrum* possui uma estrutura que proporciona o processo de inovação. O segundo, é que no *Scrum*, fornece o poder de voz as pessoas envolvidas, diferentemente de outras metodologias que, o gerente requisitava o que estava no requisito, o time analisava o que foi solicitado e literalmente entregava o que estava no requisito, sabendo que em determinado ponto, poderia ocorrer algum problema, entretanto, o time não tinha voz, logo, não eram ouvidos, sendo uma estrutura bastante ruim, quando se pretende aumentar o nível de criatividade e apresentar novos produtos e serviços no mercado, pois, as pessoas se sentem blindadas e perdem este benefício humano de contribuição, gerando uma sensação de empoderamento nos envolvidos; já em relação ao terceiro ponto, o EL acrescenta que o perfil das pessoas envolvidas, também é muito importante, pois, além das habilidades técnicas envolvidas nos processos, com o *Scrum*, ele exige ainda mais da contribuição de cada indivíduo que esteja atuando, é por isso que, as pessoas saem de um comportamento robótico e passam a agir de forma natural.

Além dos pontos apresentados, o DEV acrescenta que os *frameworks* ágeis, presam mais pela comunicação, do que a documentação, como é um dos focos do cascata, motivo da existência da figura do *Scrum master* e do *product owner*; justifica também, que é muito mais fácil ouvir do cliente o que ele espera e precisa, do que ficar aguardando a conclusão de algum documento que não remete fielmente as expectativas do cliente. Segundo o DEV, a comunicação pessoal com o cliente é muito mais eficaz, do que uma documentação, que, em algum momento, pode ocorrer de não estar completa ou equivocada.

Com relação às alterações e causas após o uso do *Scrum*, o ACL relata que tudo sofre alteração, não sendo possível realizar comparações, mas fica evidente a melhora nos processos; com relação ao cliente, ele se envolve mais com o projeto, o time passa a atuar de forma mais efetiva, as pessoas começam a entender o propósito daquilo que estão fazendo, melhora a comunicação, melhora a inovação por existir uma maior segurança por parte da equipe, os testes passam a ser mais efetivos e aumenta o nível de criatividade. A visão, passa a ser diferente, e aquilo que era abstrato, passa a ser concreto, todos conseguem enxergar o que está sendo desenvolvido e isso minimiza a probabilidade de erro no final, com isso, a qualidade fica elevada e os produtos passaram a não ser mais quase prontos e sim com status *done* ao final da *sprint*.

Sobre as alterações no nível de inovação, o EL relata sobre o processo de inovação, que se dá a partir da interação do time de desenvolvimento, quando as espirais não são mais definidas com o sangue, dando a liberdade de fato para os integrantes do *Scrum team* contribuir, não somente desenvolvendo, mas também, propondo soluções que possam ajudar ainda mais o andamento dos processos, e isso motiva, e passam a ter um propósito no projeto e isso gera inovação, estimula a criatividade e aumenta a qualidade. Ele acrescenta sobre este ponto, explicando que as pessoas precisam entender qual é o propósito daquilo estão fazendo, pois, caso não saibam, dificulta em conseguir motivar e estimular esta pessoa.

Segundo ele, as pessoas precisam receber autonomia, se a cada atividade que pessoa vai realizar, ela precisar pedir autorização, isso vai afetar na motivação da pessoa, e o máximo neste ponto, é conseguir fazer com que a pessoa consiga se superar, e que ela consiga enxergar aquilo que está

sendo bem feito, isso é algo sensacional, conseguir usar a sua habilidade para resolver um problema, que acaba afligindo a todos, e isso é maestria.

Ele acrescenta que além do *Scrum* proporcionar um cenário propício que estimula a motivação, vale ressaltar que, é necessário a iniciativa de cada pessoa, e isso tem a ver com a força motriz de cada um, é a chama do negócio; e conforme o time está motivado, e se tem uma boa base técnica, esse time possui condições de realizar um ótimo trabalho.

4.5 Prazo de Entrega

Sobre as maiores dificuldades enfrentadas na utilização do *Scrum*, no cumprimento do prazo de entrega, o ACL descreve que um dos problemas está relacionado no controle do que precisa ser entregue, e isso está diretamente ligado ao *tester*, por causa que, só foi feita a entrega nos últimos dias, e o *tester* acaba ficando parado até que ele recebe o que foi feito, e muitas vezes, a história não foi entregue, pois, faltou o teste.

No ponto de vista do EL, a primeira grande dificuldade está com relação da intervenção de alguma pessoa que pode mandar no projeto, como por exemplo algum diretor, CEO entre outros; quando esta figura começa a fazer alterações que altera o formato que o time está atuando, ou de alguma outra forma, se torna algo bastante nocivo.

Um dos problemas apresentados pelo SM, é com relação ao cliente em aceitar o formato e querer alterar a regra quando se depara com alguma demanda que não estava prevista e isso acaba atrapalhando no desenvolvimento das demais *sprints*, informa também que o cliente precisa entender que ele não pode alterar o fluxo das *sprints*, pois, elas serão afetadas diretamente, principalmente quando, o cliente necessita de uma análise, uma demanda qualquer ou até mesmo a alteração do escopo, isso dificulta o uso do *Scrum*.

Segundo o DEV, ele menciona que o *Scrum* atua com a incerteza, portanto, existe a dificuldade em poder alinhar, e essa incerteza normalmente não é comum no ambiente de negócios, por não saber como e quando ela irá terminar, e isso gera ansiedade no cliente e com o time, e às vezes, isso acaba atrapalhando nos processos. Por fim e complementando o que foi abordado pelo DEV, o PM afirma que é necessário tentar definir o volume de trabalho estimado, pois às vezes o time acaba se complicando com a entrega da *sprint*, sugerindo a criação de *sprints* menores, e assim, ele consegue conciliar possíveis imprevistos requisitados pelo cliente.

4.6 ROI

Com relação à capacidade de agregar e promover valor a partir do uso do *Scrum*, o ACL explica que quanto mais rápido o produto que foi criado for lançado no mercado, e se for possível averiguar se aquilo que o usuário está utilizando é o que foi planejado, isso é um dos princípios do *Scrum*, e isso agrega valor, diferentemente do modelo cascata, onde o objetivo é entregar no prazo com o custo que foi acordado, não se considera se o cliente terá algum lucro ou não, se irá funcionar ou não, a premissa é, entregar na data. E partindo da ideia de entregar valor, no *Scrum*, este conceito é permeado em todos os processos, respondendo de forma imediata a qualquer pressão do mercado.

De acordo com o EL, o *Scrum* já foi concebido para agregar valor, a partir do momento que as entregas passam a ser contínuas, reduzindo o tempo de entrega e reduzindo custos, deste modo, automaticamente está se gerando valor com o cliente. No momento que as pessoas conseguem visualizar o que está sendo feito e estão abertas para sugerir melhorias ou apresentar possíveis problemas, surgem novas ideias e soluções que irão aprimorar aquilo que já estava sendo realizado, e normalmente, profissionais que não possuem tanta experiência, às vezes, sugere alguma melhoria que nem mesmo o cliente havia pensado, e isso agrega valor, pois, a ideia que foi apresentada é simples e enxuta, mas terá uma enorme economia financeira ou no tempo de desenvolvimento. Quanto ao ROI, ele afirma que a classificação dos requisitos é algo essencial, às vezes, este requisito acaba assustando pelo tamanho que tem, mas quando é colocado para rodar, acabam identificando que eles poderiam

ser simples e bem menores, e com isso, conseguem entregar antes do prazo esperado, com um custo menor chegando ao mesmo ponto.

Na perspectiva do DEV, ele descreve que o cliente possui uma maior clareza, pelo fato dele poder acompanhar a evolução do projeto por meio do quadro Kanban; além de concordar com o que foi abordado pelo ACL e EL, quanto a figura do *product owner* atuando no projeto, pois, em determinado momento, ele possui a visão mais aferida com relação às necessidades da empresa e do projeto ao mesmo tempo, conseguindo priorizar aquilo que é mais relevante em determinados momentos, e isso pode proporcionar em diversos momentos, a melhoria do ROI por priorizar a necessidade do cliente e conseguir lançar o produto no momento mais oportuno.

5. Conclusão

Após o levantamento bibliográfico e das análises nas questões apresentadas, o *framework Scrum*, é realmente simples, ágil e corresponde com as mudanças e exigências do mercado atual; é normalmente sugerido no desenvolvimento de *software* em cenários não preditivos, contribuindo com uma estrutura que promove a inovação em cada ciclo de desenvolvimento e na superação das expectativas por parte dos clientes, embora, requer a constante presença do cliente nas decisões e suporte ao time de desenvolvimento no decorrer do projeto.

Pode se chegar ao objetivo deste trabalho, através das soluções e experiências apresentadas pelos profissionais da área, com o uso do *Scrum*; quais foram os benefícios proporcionados com o seu uso e também, através das recomendações por parte dos participantes, que agregaram valor neste trabalho e possibilitou um melhor entendimento quanto as possíveis soluções em relação aos problemas vivenciados nesta área, como nos casos apresentados pelo SM e ACL, que relataram uma maior aceitação pelo cliente no uso do *Scrum* quando, ele já possui a cultura organizacional da empresa, com o *mindset* voltado para o *Lean* ou com uma cultura voltada aos princípios do *Scrum*, e isso contribuí para o processo de inovação e no desenvolvimento do produto final.

Com relação ao envolvimento da figura do cliente no processo de desenvolvimento do projeto, isso foi apresentado com um fator decisivo e de grande impacto no resultado final e de soluções no decorrer do projeto, no qual o time de desenvolvimento, muitas vezes não consegue mensurar o que de fato foi solicitado, e com a presença do cliente, esse ponto é rapidamente solucionado e de forma simples, rápida e eficaz. Em contrapartida, quando isso não acontece, o cliente é a parte mais afetada e acaba perdendo a oportunidade de agregar valor no momento mais oportuno do projeto, podendo minimizar eventuais custos, reduzindo o tempo de entrega do produto ao mercado.

Vale ressaltar que, quanto menor for o nível de burocratização da empresa e quanto menos ela for hierarquizada, melhor é a aceitação no uso do *Scrum*; quanto mais existir a figura do líder ou a figura do chefe no processo de inovação e de soluções no processo de desenvolvimento, pior será o resultado esperado. Por isso é necessário analisar bem o resultado esperado, antes de iniciar um processo ou utilização de um método, para que ao seu término, atenda as expectativas dos *stakeholders* envolvidos e também as necessidades do mercado atual.

É notório que o *Scrum* corresponde de forma ágil e eficaz as pressões e necessidades do mercado, mas é necessário avaliar em qual nível de maturidade a cultura organizacional da empresa contratante está, ao implantar um *framework* em seus projetos, pois, a sua performance é realmente atrativa em cenários imprevisíveis.

Conclui-se que, este *framework* revolucionou a forma que as empresas atuam em seus projetos e no desenvolvimento de *software*, e isso contribuí para o processo de inovação e interação entre a equipe, além da melhoria na comunicação com o cliente, aumenta o nível de qualidade e de valor agregado ao produto final, diferentemente de outros métodos tradicionais que acabam não proporcionando tais interações e o produto final, acaba não agregando valor. Sugere-se para novos estudos, a continuidade deste estudo, buscando identificar a percepção das empresas contratantes, quanto aos métodos tradicionais x o uso de *frameworks*, como no caso do *Scrum*, a fim de se obter,

um entendimento maior em relação à figura do líder, do gerente de projeto e a figura do chefe no processo de inovação e soluções em *software* para a empresa.

Referências Bibliográficas

- Agile Learning Labs: **The Agile Dictionary**, Disponível: <http://agiledictionary.com/207/Scrum/>
- Amaral, D. C. & Conforto, E. C. **Métodos ágeis para o gerenciamento de projetos**. In: Jugend, D., Barbalho, S. C. M. & Silva, S. L. (Org.). **Gestão de Projetos: teoria, prática e tendências**. Rio de Janeiro: Campus, p. 183-207.2014.
- Bassi, D. L., Filho **Experiências com Desenvolvimento Ágil**. Dissertação de Mestrado. Instituto de Matemática e Estatística da Universidade de São Paulo. 150p. 2008. Disponível: <https://www.ime.usp.br/~dairton/files/Dissertacao-DairtonBassi.pdf>
- Beck, K., Cunningham, W., Hunt, A., Martin, R.C, Thomas, D., Beedle, M., Fowler, M., Jeffries, R., Mellor, S., Bennekum, A. V., Grenning, J., Kern, J., Schwaber, K. Cockburn, A., Highsmith, J., Marick, B., Sutherland, J. (2001). **Manifesto for Agile Software Development**. Recuperado em 12 outubro 2016, de <http://agilemanifesto.org/>
- Boehm, B., 2006, **A View of 20th and 21st Century Software Engineering**. In: **Proceedings of the 28th International Conference on Software Engineering (ICSE)**, Shangai, China, pp.12-29.
- Brod, C. (2013) **ScrumScrum Guia Prático para Projetos Ágeis**. (2a. ed.) São Paulo: Novatec editora.
- Carvalho, B. V. & Mello, C. H. P. (2012) **Aplicação do Método Ágil Scrum no Desenvolvimento de Produtos de Softwares em uma Pequena Empresa de Base Tecnológica**. *Gestão & Produção*, 19(3), 557-573.
- Cooper, D. & Schindler, P. (2004). **Método de Pesquisa em Administração**. Porto Alegre: Bookman.
- Date, R. N., Pinochet, L.H.C., Bueno, R.L.P. & Nemoto, M.C.M.O, 2016. **Aplicação do método ágil Scrum em uma fundação educacional do setor público**. *Revista de Gestão e Projetos*, v. 7, n. 2.
- Denning, S. (2013). **Why Agile can be a game changer for managing continuous innovation in many industries**. *Strategy & Leadership*, Chicago, v. 41, n. 2, p. 5-11.
- Dingsoyr, T., Nerur, S., Balijepally, V. & Moe, N. B. (2012). **A decade of agile methodologies: Towards explaining agile software development**. *Journal of Systems and Software*, 85(6), 1213–1221. Disponível: <http://doi.org/10.1016/j.jss.2012.02.033>
- Duane, P. T., Richard, B. & Heinz, K. (1999) **Growing systems in emergent organisations**, *Commun. ACM*, Vol 42 (8), pg117-123.
- Gil, A. C. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas, 2009. 175 p.
- Heitor, P. J. E. S. (2017) **Aplicação do modelo Scrum ao processo de desenvolvimento de software da cgd**. Dissertação de Mestrado. Instituto Universitário de Lisboa. 99p. 2017. Disponível:<https://repositorio.iscte-iul.pt/bitstream/10071/11650/1/Tese%20Paulo%20Heitor.pdf>
- Machado, M. & Medina, S. G. (2009). **Scrum – Método Ágil: Uma Mudança Cultural na Gestão de Projetos de Desenvolvimento de Software**. *Revista Científica Intraciência*, Faculdade do Guarujá – UNIEESP, 1(1), 58-71
- Minayo, M.C.S. (2001) **Pesquisa Social. Teoria, método e criatividade**. 18 ed. Petrópolis: Vozes, 2001.
- Sabbagh, R. (2013) **Scrum: Gestão Ágil Para Projetos De Sucesso**. 1º edição. São Paulo: Casa do Código, 2013.
- Schwaber, K. (2004). **Agile Project Management with Scrum**. EUA: Microsoft Press. 2004.
- Schwaber, K. & Sutherland, J., (2017). **Guia Do Scrum** Disponível: <http://www.Scrumguides.org/docs/Scrumguide/v2017/2017-Scrum-GuidePortuguese-Brazilian.pdf>
- Silva, E. C., & Lovato L. A. (2016). **Framework Scrum: Eficiência Em Projetos De Software**. *Revista de Gestão e Projetos*, 7(2), 1-15.
- Takeuchi, Hirotaka & Nonaka, (1986), **New Product Development Game**. *Harvard Business Review* 64, no. 1 (January–February 1986).

Vieira, D. (2014) **ScrumScrum: A Metodologia Ágil Explicada de forma Definitiva.** **MindMaster.** Disponível: <http://www.mindmaster.com.br/Scrum>