

PROPOSTA DE COMPARAÇÃO ENTRE O FLUXO DA METODOLOGIA LEAN E SCRUM: CENÁRIOS POSITIVOS E NEGATIVOS NAS ORGANIZAÇÕES

*PROPOSAL FOR COMPARISON BETWEEN LEAN AND SCRUM METHODOLOGY
FLOW: POSITIVE AND NEGATIVE SCENARIOS IN ORGANIZATIONS*

Vanessa Coelho Ribeiro,
Renato José da Silva Camões,
Patrícia Lúcia de Castro Cerqueira

RESUMO

As metodologias ágeis utilizadas em desenvolvimento de software quebram o paradigma do desenvolvimento em cascata e outros processos mais tradicionais. Porém elas não são uma substituição aos processos já existentes, mas sim uma complementação ou uma alternativa. Esse artigo apresenta uma comparação do fluxo das metodologias ágeis, mais especificadamente os métodos Scrum e Lean Software Development (LSD). O objetivo é apresentar um estudo de caso através da análise comparativa e demonstrar os pontos incomuns e as diferenças. Também será apresentado o cenário atual das organizações em relação a fatores bem consolidados e fatores que precisam de mais empenho.

INTRODUÇÃO

A partir dos avanços tecnológicos da academia e da indústria, o software se tornou uma ferramenta indispensável em todas as atividades da sociedade contemporânea. Cada vez são produzidos novos equipamentos eletrônicos de forma mais rápida e a crescente expansão destes dispositivos tem como consequência a determinação de um desenvolvimento de software cada vez mais especializado e avançado. Isto resulta diretamente no aumento da necessidade por softwares e causa de uma forma indireta na busca por maneiras mais apropriadas de desenvolvê-los. Partindo deste princípio, o mercado de software tem se tornado um tanto competitivo, fazendo com que a Engenharia de Software desenvolva a capacidade de se adaptar a estas mudanças frequentes (BASSI FILHO, 2010).

A princípio, a indústria de desenvolvimento de software seguiu uma metodologia tradicional, contendo as fases de levantamento e análise de requisitos, desenho da arquitetura, implementação, testes, produção e manutenção. Embora

correta, essa metodologia é vista como desnecessariamente rígida, ao tornar o processo tão formal, ela impede que os clientes tenham o desenvolvimento na velocidade que necessitam. Além disso, muitas empresas ainda utilizam a metodologia tradicional, um exemplo é o modelo em Cascata, que é composto basicamente por atividades sequenciais de levantamento de requisitos, análise, projeto, implementação, teste, implantação e manutenção. Este modelo é derivado de outras engenharias tradicionais (Civil, Elétrica, Naval...) e foi o primeiro a ser usado pela Engenharia de Software, na década de 70 (PRESSMAN, 2001).

Oposto a essas metodologias tradicionais, em 2001, um grupo de programadores lançou o Manifesto Ágil, pregando uma tecnologia que tem o objetivo de satisfazer os clientes entregando com rapidez e com maior frequência versões do software conforme as necessidades, contando com quatro valores principais. Ou seja, pode-se evoluir o software de acordo com as necessidades do cliente e das demandas da sociedade, do contrário, o produto final demoraria tanto para ficar pronto que poderia se tornar obsoleto (CAMARGO, 2018).

Desta forma, nota-se que o mercado esta sempre se reinventando, para as necessidades do desenvolvimento de novas soluções, a fim de encontrar forma de potencializar resultados no ambiente empresarial. Uma das soluções é o framework crum, uma das ferramentas ágeis mais utilizada no mundo por equipes de desenvolvimento de software, levando à compreensão de possibilidade de aplicação a qualquer projeto. Uma outra é a metodologia Lean. A agilidade promovida por ela está relacionada com o enxugamento de processos. É um método bastante utilizado em startups e gestão de algumas empresas que estão começando, isso porque, normalmente, o desperdício para elas são mais do que simplesmente prejudiciais.

O objetivo deste trabalho é realizar um estudo de caso através de análise bibliográfica, aplicação da técnica de questionário em relação a 2 metodologias ágeis e avaliar como essas metodologias auxiliam no desenvolvimento de software, bem como, realizar uma análise comparativa entre as 2 metodologias e suas principais características, destacando as mais relevante ao processo de desenvolvimento de software.

MANIFESTO ÁGIL

De acordo com o site metodoagil.com, o manifesto ágil é uma declaração de valores e princípios essenciais para o desenvolvimento de software, ele foi criado

em fevereiro de 2001, onde um encontro entre 17 líderes que trabalhavam no contra-fluxo dos padrões da indústria de software, objetivando chegar a uma nova metodologia de produção de software, que pudesse ser usada por todos eles e em outras empresas, substituindo os modelos tradicionais de desenvolvimento. O grupo chegou ao consenso de que alguns princípios eram determinantes para a obtenção de bons resultados. (LEAN, 2019)

O resultado deste encontro foi à identificação de 12 princípios e a publicação do Manifesto Ágil que os representa com quatro premissas:

- Indivíduos e iterações são mais importantes do que processos e ferramentas;
- Software funcionando é mais importante do que documentação completa;
- Colaboração com o cliente é mais importante do que negociação de contratos;
- Adaptação a mudanças é mais importante do que seguir o plano inicial;

1	Prioridade é satisfazer o cliente através da entrega contínua e adiantada de software com valor agregado;
2	As mudanças nos requisitos são bem-vindas, mesmo tardiamente no desenvolvimento. Processos ágeis tiram vantagem das mudanças visando vantagem competitiva para o cliente;
3	Frequentes entregas do software funcionando, de poucas semanas a poucos meses, com preferência à menor escala de tempo;
4	As pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto;
5	Os projetos devem ser construídos em torno de indivíduos motivados. Dando o ambiente e o suporte necessário e confiança para fazer o trabalho;
6	O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é através de conversa face a face;
7	Software funcionando é a medida primária de progresso;
8	Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente;
9	Contínua atenção a excelência técnica e bom design aumentam a agilidade;
10	Simplicidade para maximizar, a quantidade de trabalho não realizado é essencial;
11	As melhores arquiteturas, requisitos e designs emergem de equipes autoorganizáveis;
12	Em intervalos regulares, a equipe deve refletir sobre como tornar-se mais efetiva, e então, ajustar-se de acordo com seu comportamento.

Tabela 1. Princípios do Manifesto Ágil
(KALERMO E RISSANEN 2002)

O Manifesto Ágil ressalta o que mais tem valor para as metodologias ágeis, a importância de como saber lidar com pessoas, assim como ter o cliente colaborando

para encontrar a melhor solução, entregar o software com qualidade e do que se adaptar às mudanças. Isto oculta parte das dificuldades dos processos, contratos, documentação e planejamento para o desenvolvimento de um software. (BASSI FILHO, 2008)

SCRUM

De acordo com o site desenvolvimentoagil.com.br, o Scrum é uma metodologia ágil e iterativa para gestão e planejamento de projetos de software, pois o trabalho é dividido em iterações que são chamadas de Sprint, os projetos são divididos em ciclos (mensais) chamados de Sprints no caso do Scrum. O Sprint representa um Time Box dentro do qual um conjunto de atividades deve ser executado.

Ainda de acordo com o site, as funcionalidades a serem implementadas em um projeto são mantidas em uma lista que é conhecida como Product Backlog. No início de cada Sprint, faz-se um Sprint Planning Meeting, ou seja, uma reunião de planejamento na qual o Product Owner prioriza os itens do Product Backlog e a equipe seleciona as atividades que elas será capaz de implementar durante o Sprint que se inicia. As tarefas alocadas em um Sprint são transferidas do Product Backlog para o Sprint Backlog.

Abaixo é apresentada uma imagem muito utilizada para representar as interações entre as atividades no processo.

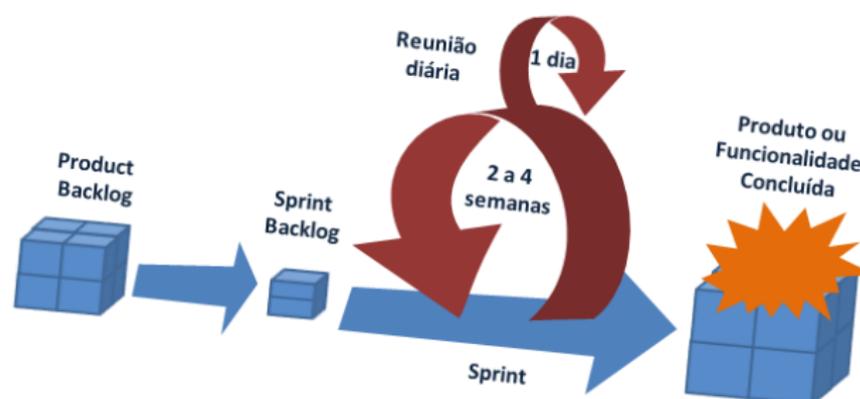


Figura 1. Interações entre as atividades no processo.
(BALDO, 2019)

De acordo com o site mindmaster.com é válido ressaltar os papéis fundamentais. Product Owner é o ponto central com poderes de liderança sobre o produto, é o único responsável por decidir quais recursos e funcionalidades serão construídas e qual a ordem que devem ser feitos. Porém, é responsabilidade dele manter e comunicar a todos os outros participantes uma visão clara do que a equipe Scrum está buscando alcançar no projeto. Como tal, ele é responsável pelo sucesso global da solução. Para garantir que a equipe construa rapidamente o que o Product Owner precisa, ele deve colaborar ativamente com o ScrumMaster e equipe de desenvolvimento e deve estar disponível para responder às perguntas tão logo estas são feitas.

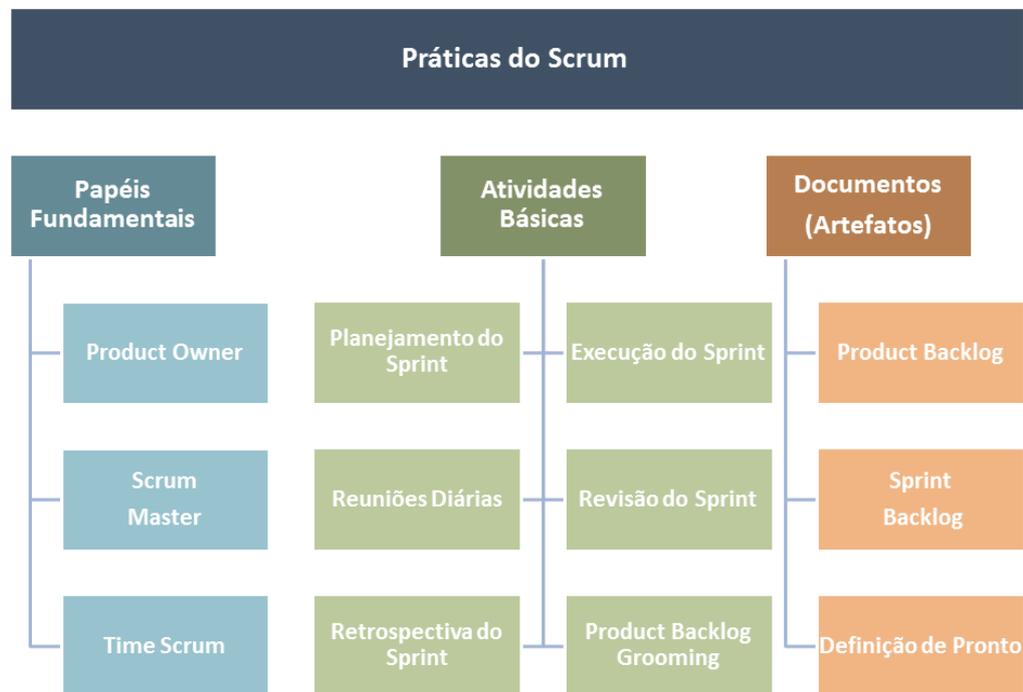


Figura 2. Práticas do scrum (SCHWABER, 2019)

O Scrum Master procura assegurar que a equipe respeite e siga os valores e as práticas do Scrum, ele também protege a equipe assegurando que ela não se comprometa excessivamente com relação àquilo que é capaz de realizar durante um Sprint. O Scrum Master atua como facilitador do Daily Scrum e torna-se responsável por remover quaisquer obstáculos que sejam levantados pela equipe durante essas reuniões. O papel de Scrum Master é tipicamente exercido por um gerente de projeto ou um líder técnico, mas em princípio pode ser qualquer pessoa da equipe. (CRUZ, 2019)

O Scrum Team é a equipe de desenvolvimento e tem de 6 a 10 pessoas, nela não existe uma divisão funcional através de papéis tradicionais, tais como programador, designer, analista de testes ou arquiteto. Todos no projeto trabalham juntos para completar o conjunto de trabalho com o qual se comprometeram conjuntamente para um Sprint.

Abaixo é apresentada uma tabela explicando as atividades do processo Scrum:

Product Backlog	É uma lista contendo todas as funcionalidades desejadas para um produto, o conteúdo desta lista é definido pelo Product Owner.
Sprint Backlog	É uma lista de tarefas onde os itens são extraídos do Product Backlog, que o Scrum Team se compromete a fazer em um Sprint.
Daily Scrum	Reunião diária feita a cada dia do Sprint, para disseminar conhecimento sobre o que foi feito no dia anterior, identifica impedimentos e priorizar o trabalho a ser realizado no dia que se inicia.
Sprint Planning Meeting	Reunião de planejamento que estão presentes o Product Owner, o Scrum Master e todo o Scrum Team, onde o Product Owner descreve as funcionalidades de maior prioridade para a equipe.
Sprint Review	Reunião feita ao final de cada Sprint, o Scrum Team mostra o que foi alcançado durante o Sprint, o projeto é avaliado em relação aos objetivos do Sprint, determinados durante o Sprint Planning.
Sprint Retrospective	Ocorre ao final de um Sprint e serve para identificar o que funcionou bem, o que pode ser melhorado e que ações serão tomadas para melhorar.
Release Burndown Chart	A equipe monitora seu progresso em relação a um plano atualizando um Release Burndown Chart ao final de cada Sprint(iteração). O eixo horizontal mostra os Sprints e o eixo vertical mostra a quantidade de trabalho que ainda precisa ser feita no início de cada Sprint.

Tabela 2. Atividades do processo scrum (CRUZ, 2019)

Ainda que resumidamente, percebe-se que o Scrum foi criado exatamente para o gerenciamento e planejamento de projetos de desenvolvimento de softwares. Trata-se de uma estrutura de processos organizados, envolvendo seus executores e serviços. Hoje é uma das ferramentas mais conhecida para desenvolvimento ágil, porém ainda existem outros tipos de métodos ágeis que podem ser aplicados em um projeto.

LEAN SOFTWARE DEVELOPMENT (LSD)

O Sistema Toyota de Produção, também conhecido como Lean Manufacturing, surgiu no Japão, na fábrica de automóvel Toyota, logo após a Segunda Guerra Mundial. Nesta época a indústria japonesa tinha uma produtividade muito baixa e uma enorme falta de recursos, o que naturalmente a impedia adotar o modelo de produção em massa. (FADEL, 2019)

Segundo o Lean Institute Brasil, a metodologia Lean é uma estratégia de negócios que busca aumentar a satisfação do cliente através de um melhor aproveitamento dos recursos. A gestão Lean busca oferecer aos clientes um valor com o custo mais baixo dos seus produtos (propósito) através de melhorias contínuas dos seus fluxos de valor primário e de suporte (processos) através de pessoas com iniciativa, motivadas e qualificadas (pessoas). A implementação desta metodologia deve se encontrar nas reais necessidades dos negócios da empresa e não na simples aplicação das ferramentas lean. Ou seja, a metodologia Lean busca atender a necessidade do cliente da maneira mais simples possível, com um menor valor, aproveitando ao máximo todos os recursos disponíveis para a produção e tendo como consequência um melhor custo benefício para o cliente.

Segundo os autores Aline Fadel e Henrique Silveira (2010), a metodologia Lean é distribuída em sete princípios: eliminar o desperdício, amplificar o aprendizado, adiar comprometerimentos e manter a flexibilidade, entregar rápido, tornar a equipe responsável, construir integridade e visualizar o todo.

Eliminar o desperdício	Tudo aquilo que não agrega valor para o cliente final e que não são percebidos pelo cliente é considerado desperdício.
Amplificar o aprendizado	Priorizar a comunicação e o feedback contínuos entre equipes e usuários durante o processo de desenvolvimento de software.
Adiar Comprometimentos e manter a flexibilidade	Deixar as decisões e comprometerimentos para o último momento responsável, permitindo coletar informações e ter experiências para fortalecer a tomada de decisão.
Entregar rápido	Entregar o software de valor de forma rápida e contínua.
Tornar a equipe responsável	Criar um ambiente onde a equipe trabalhe de forma auto-organizada e auto-dirigida, evitando micro-gerenciamento.
Construir Integridade	Garantir a qualidade no desenvolvimento do software utilizando técnicas de teste unitário, refatoração e integração contínua.

Visualizar o todo	Entender que o software concluído é muito mais que a soma das partes entregues e verificar como ele está alinhado com os objetivos da empresa.
--------------------------	--

Tabela 3. Princípios da metodologia Lean Fadel e Silveira (2010)

Diante dos princípios da metodologia Lean, percebe-se que seu objetivo é de eliminar o desperdício, desfazer dos componentes que não possuem valor agregado no processo. Seguindo os princípios pode se criar melhorias em eficiência, levando a custos mais baixos e uma maior competitividade, melhorando a forma como uma equipe trabalha em conjunto, o gerenciamento de inventário e até mesmo a interação do cliente.

METODOLOGIA

Como metodologia foi utilizada o questionário que é um instrumento de coleta de dados, constituído por uma série ordenada de perguntas, que serve para elaborar pesquisas específicas de acompanhamento de certo assunto com usuários selecionados podendo ser enviado a diversos grupos que podem estar em diversos locais diferentes.

Para realização do estudo, foi enviado um questionário com 10 perguntas para profissionais na área de TI (desenvolvimento, infraestrutura, banco de dados, gestão, etc), com portes e organizações variados.

No gráfico 01, conforme a pesquisa 50% é de empresa de grande porte, ou seja, 17 dos entrevistados realizam suas atividades em empresas de grande porte, esse resultado foi bastante significativo na realização desse trabalho.

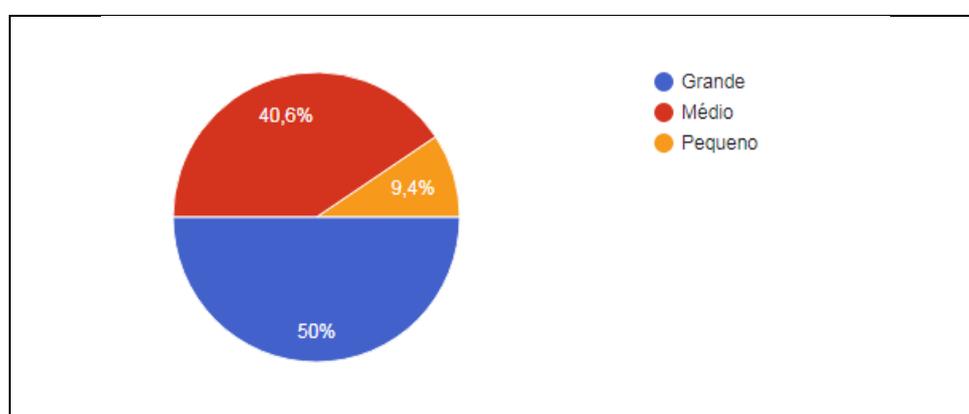


Gráfico 1. Percentual do porte das empresas das pessoas que participaram da pesquisa (AUTOR, 2019)

No gráfico 02, conforme pesquisa 47,1% dos entrevistados trabalham na área de desenvolvimento.

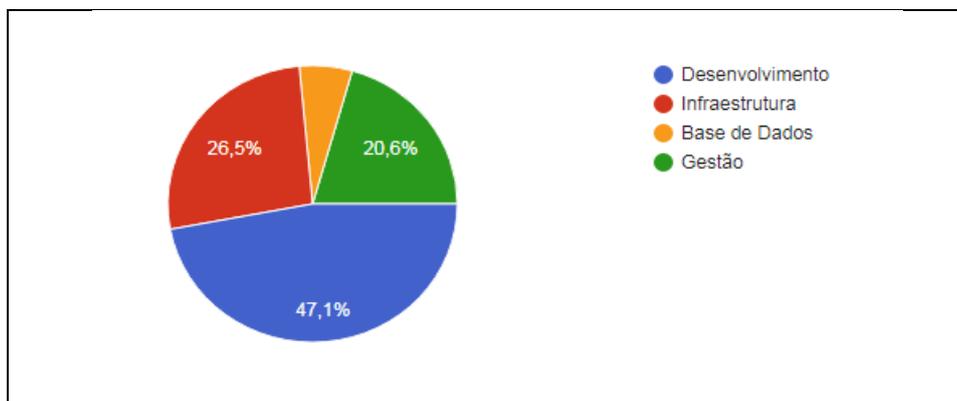


Gráfico 2. Percentual da área de TI aonde os entrevistados atuam (AUTOR, 2019)

É importante ressaltar nesse cenário 2 que o manifesto ágil acaba impactando outras equipes e que seus valores, tal como, indivíduos e interações mais que processos e ferramentas, adaptam equipes de não desenvolvedores para repensar a gestão de seus projetos. É uma maneira de enxergar o trabalho e realização de tarefas que podem, sim, trazer muitos benefícios para equipes de todas as áreas.

ANÁLISES E RESULTADOS

A tabela abaixo mostra as perguntas, o percentual e os impactos relacionados à pesquisa feita, que visa analisar o estudo do tema de metodologias ágeis observado na realidade e explicar como e porque ele ocorre, identificando os fatores que contribuem para que o tema em questão se materialize.

Ponto de análise	Percentual	Impactos
1-O uso de um processo definido para desenvolvimento de software engessa o aprendizado?	56,3%	18 entrevistados dizem que frequentemente o uso de um processo definido engessa o aprendizado. Diante da pesquisa esse resultado mostra que quase metade das empresas estão inseguras em adotar ou não adotam um tipo de metodologia ágil que visa o aprendizado. Esse aprendizado pode ser composto por um ciclo conforme o SCRUM, que identifica o problema, sua causa, uma solução, a implementação, correções e o novo cenário.
2-Adiar decisões permite que as escolhas sejam apoiadas por mais experiência e conhecimentos	53,1%	17 entrevistados dizem que frequentemente adiar decisões permite escolhas mais experientes, mas, muitas vezes adiar decisões e causada por medo de tomar decisões equivocadas e não saber o que fazer diante da situação. Na verdade FILHO (2008) relata que nessa situação a equipe deve ter

adquiridos no decorrer do processo de desenvolvimento de software?		a capacidade de absorver mudanças tratando os planejamentos como estratégias para atingir um objetivo e não como comprometimentos. Percebe-se que as respostas (mesmo que de forma equivocada na interpretação) mantêm um cenário positivo para mais da metade das organizações entrevistadas.
3-A velocidade de desenvolvimento também ajuda atender às necessidades atuais do cliente?	53.1%	17 entrevistas dizem que frequentemente a velocidade de desenvolvimento atende as necessidades do cliente, o que faz desse ponto também positivo. Essa inovação de maneira rápida pode acabar comprometendo a qualidade que os clientes tanto demandam. Porém a velocidade de desenvolvimento também ajuda atender às necessidades atuais do cliente, permitindo que possam ser dadas outras decisões (por meio da entrega rápida), e entregando um produto com um bom nível de acerto por meio do acúmulo de conhecimento.
4-Envolver os desenvolvedores nas decisões de detalhes técnicos é fundamental para atingir a excelência no desenvolvimento de software?	50%	16 entrevistados dizem que frequentemente com o envolvimento de desenvolvedor nas decisões se atingi uma excelência do desenvolvimento. Ainda que isso represente metade das organizações, esse cenário é muito ruim porque muitas organizações ainda não levam em consideração o conhecimento dos detalhes técnicos, que deve ser levado em consideração na tomada de decisões, na definição de processos e conseqüentemente na entrega do melhor produto.
5-Em sua opinião deve-se gastar tempo para encontrar e corrigir defeitos, ou deve-se investir na prevenção através de vários tipos de teste.	90,3%	28 entrevistados concordam que deve se investir na prevenção através de vários tipos de teste. Nesse caso percebe-se que a grande maioria das organizações já adotam testes automatizados e preservando a flexibilidade para mudanças, adaptações e extensões trazendo segurança e motivação para alcançar níveis mais elevados de qualidade com relata FILHO (2008).
6-Em sua opinião, no desenvolvimento de software vale mais: Indivíduos e suas interações ou processos e ferramentas?	75%	24 entrevistados concordam que indivíduos e suas interações, valem mais no desenvolvimento de software. Nesse caso numa empresa indivíduos são mais importantes, sem eles é impossível dela funcionar, mas é preciso também uma boa iteração. Esse cenário é positivo perante a pesquisa e aponta que as organizações buscam soluções mais integradas para prover bons resultados no desenvolvimento e implantação de software.

7-Em sua opinião o que deixa o desenvolvimento de software mais assertivo: Colaboração com o cliente ou negociação de contratos?	84,4%	27 entrevistados concordam que com a colaboração Singh, Mona. "U-SCRUM: An agile methodology for promoting usability." <i>Agile 2008 Conference</i> . IEEE, 2008.com o cliente o desenvolvimento de software fica mais assertivo. Nesse caso à uma relação colaborativa entre cliente e desenvolvedor para produzir um software que traga valor ao cliente. Deve-se trabalhar em conjunto com o cliente e não contra ele ou ele contra a equipe de desenvolvimento.
8-Em sua opinião seguir planos de desenvolvimento de software responde a mudanças de forma efetiva?	56,3%	18 entrevistas dizem que frequentemente seguir planos responde a mudanças efetiva, no entanto um plano de projeto deve ser flexível o suficiente para ser mudado, deve haver espaço para mudá-lo à medida que sua situação muda, caso contrário, seu plano rapidamente se tornará irrelevante, seus processos poderão já não ser mais efetivos e talvez sua equipe não seja mais necessária para o cliente/fornecedor. Por isso esse ponto também se tornou negativo nessa pesquisa. Singh, Mona. "U-SCRUM: An agile methodology for promoting usability." <i>Agile 2008 Conference</i> . IEEE, 2008.

DISCUSSÃO

Dois pontos são necessários para elucidar essa discussão. Em primeiro lugar deve-se destacar que o Scrum é uma metodologia ágil para gestão de projetos, que pode ser usado no desenvolvimento de software. Já o Lean tem como objetivo eliminar desperdícios e promover uma busca constante por melhoria contínua.

Porém, apesar das diferenças de objetivos, é possível verificar por meio da tabela abaixo aonde se encaixa os princípios do LEAN dentro do ciclo do Scrum.

CICLO DO SCRUM	PRINCÍPIOS DO LEAN QUE PODEM SER ATINGIDOS POR ESSE CICLO
Produto Backlog	- eliminar o desperdício - visualizar o todo
Daily Scrum (reunião diária)	-visualizar o todo - amplificar o aprendizado - adiar compromentimentos e manter a flexibilidade,
Release Burndown Chart	- visualizar o todo
Scrum Team (desenvolvimento)	- entregar rápido - eliminar o desperdício - construir integridade - visualizar o todo
Sprint Backlog	- eliminar o desperdício -visualizar o todo

Sprint Planning	<ul style="list-style-type: none"> - amplificar o aprendizado - adiar comprometerimentos e manter a flexibilidade - visualizar o todo
Sprint Retrospective	<ul style="list-style-type: none"> - amplificar o aprendizado - visualizar o todo - adiar comprometerimentos e manter a flexibilidade - construir integridade
Sprint Review	<ul style="list-style-type: none"> - amplificar o aprendizado - visualizar o todo - entregar rápido
Product Owner	<ul style="list-style-type: none"> - adiar comprometerimentos e manter a flexibilidade - amplificar o aprendizado - tornar a equipe responsável - construir integridade - visualizar o todo
Scrum Master	<ul style="list-style-type: none"> - adiar comprometerimentos e manter a flexibilidade - amplificar o aprendizado - tornar a equipe responsável - visualizar o todo

Tabela 4. Princípios do Lean dentro do ciclo do Scrum (AUTOR, 2019)

A tabela tenta unir uma filosofia a um conjunto de princípios de desenvolvimento para melhorar as entregas e os produtos alcançados. Por isso, na aplicação da metodologia Scrum pode-se se ter de forma abstrata algumas características da LEAN. Verifica-se ainda, de forma comprovada nas várias pesquisas, que ambas as metodologias ágeis têm por finalidade maximizar o trabalho das equipes de projetos e os resultados gerados aos clientes.

O segundo ponto de discussão trata-se dos resultados da pesquisa. Tem-se respostas que demonstram cenários positivos para o caminho do movimento ágil, mas também obteve-se respostas que mostram que as organizações ainda não estão preparadas para a metodologia ágil.

Como primeiro cenário negativo tem-se o uso de processos definidos como engessados, como por exemplo a forma de atendimento ao cliente, os tipos de ferramentas usadas na organização e o processo de admissão de pessoas. Tudo isso impacta no fluxo de implantação de metodologias ágeis.

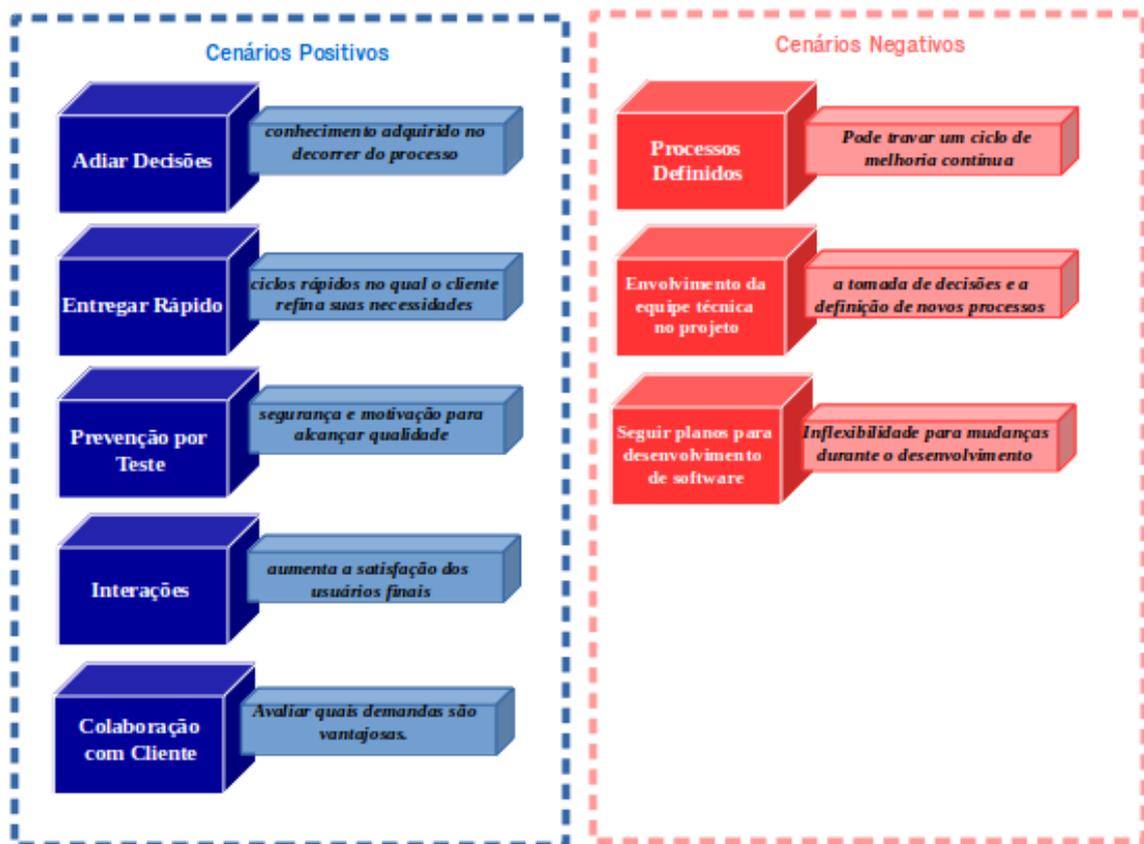


Figura 3. Cenário positivo e negativo em relação à pesquisa (AUTOR, 2019)

O segundo ponto trata-se do pouco envolvimento da equipe, ou ainda, um fluxo hierárquico da comunicação gera as informações imprecisas, não concisas, muitas das vezes sem abordar os aspectos relevantes, sendo insuficiente para causar dúvidas quanto ao objetivo.

Por fim, o terceiro ponto que colide com a implantação de metodologias ágeis são os planos que não podem ser alterados. Desta forma os profissionais responsáveis pela arquitetura de software têm a obrigação de seguir o que foi planejado. Tornam-se profissionais inflexíveis e percorrem um caminho sem buscar alternativas para projetar sistemas com a maior flexibilidade.

Antagônico a esses três pontos é percebe-se que diante do percentual da pesquisa, a maioria dos entrevistados investem na prevenção através de vários tipos de teste. Eles demonstram que os testes em software são uma maneira de garantir a qualidade do sistema e que devam ser utilizadas ferramentas que automatizam a execução, para que testes possam ser realizados de modo mais rápido e com maior abrangência.

Os entrevistados entendem que o desenvolvimento de software é uma atividade humana e que a qualidade da interação entre as pessoas pode resolver

problemas crônicos de comunicação, processos e ferramentas são importantes, mas devem ser simples e úteis. Se uma empresa jogar fora todos os seus processos e ferramentas e deixar os funcionários, ela ainda tem chances de funcionar, agora se abrir mão de seus funcionários e ficar com seus processos e ferramentas, impossível dela funcionar. Desta forma, demonstram que os indivíduos e iteração são mais importantes que processos e ferramentas, desde que todos ganhem com isso.

Outro ponto já consistente nas organizações é o fato de a colaboração com cliente deixa o desenvolvimento de software mais assertivo, pois, o cliente e os desenvolvedores estão do mesmo lado colaborando para produzir o software que traga valores para o cliente, ambos são necessários para produzir software de boa qualidade, no desenvolvimento de software a relação cliente-desenvolvedor deve ser colaborativa, ao invés de disputas de contrato.

CONCLUSÃO

Considerando que as metodologias ágeis são iterativas e incrementais, se resulta em um produto desenvolvido com base na melhoria contínua, e como o cliente participa de todo o projeto, a sua satisfação normalmente é garantida. Já é aplicada em muitas empresas do mundo e pouco a pouco demonstra suas vantagens em longo prazo.

Este trabalho conclui que as metodologias ágeis apresentadas neste poderiam ser aplicadas de maneiras complementar ou alternativa às metodologias tradicionais. Conclui-se também que na aplicação da metodologia Scrum pode-se ter de forma abstrata algumas características da Lean. O Scrum é uma metodologia ágil para gestão de projetos, que pode ser usado no desenvolvimento de software de projetos. Já o Lean tem como objetivo eliminar desperdícios e promover uma busca constante por melhoria contínua.

Como resultados positivos pode se dizer que com a implementação da metodologia ágil pode se garantir estratégias de entrega de valor mais rápida ao cliente.

Já os resultados negativos têm o uso de processo definido engessado que pode travar o fluxo de implantação de modelos ágeis tornando empresas inseguras em adotar um tipo de metodologia ágil, o que impacta no aprendizado, na flexibilidade para mudanças durante o desenvolvimento, no plano de projeto flexível que pode representar espaço para mudá-lo à medida que sua situação muda, caso

contrário, o plano se tornará irrelevante. Ressalta-se o envolvimento da equipe técnica no projeto pois muitas organizações ainda não levam em consideração o conhecimento dos detalhes técnicos, para a tomada de decisões, na definição de processos e conseqüentemente na entrega do melhor produto.

REFERÊNCIAS

BALDO, Rodrigo Fernando Galzerano, et al. "Aplicação da metodologia scrum em um estudo de caso de engenharia." *Revista Produção Online* 19.3 .2019.

BASSI FILHO, Dairton Luis: Experiências com desenvolvimento ágil. Instituto de Matemática e Estatística da Universidade de São Paulo. 2008.

BASSI FILHO, Dairton Luiz. Experiências com desenvolvimento Ágil. São Paulo, 2010. Disponível em: Acesso em: 20 de agosto 2012.

CAMARGO, Robson, O que realmente é a metodologia ágil? 2018 Disponível em <http://robsoncamargo.com.br/blog/o-que-e-metodologia-agil> Acesso em: 03 de setembro 2019.

CRUZ, Jossandro Rodrigues da, Luciana Schleder Gonçalves, and Ana Paula Magalhães de Abreu Giacomo. "Agile Scrum Methodology: implementation by the nurse in an educational game on safe medication management." *Revista gaucha de enfermagem* 40.SPE (2019).

FADEL, Aline Cristine. SILVEIRA, Henrique da Mota. Metodologias ágeis no contexto de desenvolvimento de software: XP, Scrum e Lean. Limeira, 2010. Disponível em: Acesso em: 21 de agosto de 2019.

FILHO, D. L. B.: Experiências com desenvolvimento ágil. Instituto de Matemática e Estatística da Universidade de São Paulo (Dissertação de Mestrado). 2008.

FRANCO, E. F.: Um modelo de gerenciamento de projetos baseado nas metodologias ágeis de desenvolvimento de software e nos princípios da produção enxuta. Escola Politécnica da Universidade de São Paulo (Dissertação de Mestrado). 2007.

HENRIQUES, Cecília Machado, et al. "SCRUM: REFLEXÕES SOBRE A APRENDIZAGEM EO USO DA ABORDAGEM PARA INOVAÇÃO DIGITAL." *International Congress of Knowledge and Innovation-Ciki*. Vol. 1. No. 1. 2018.

KALERMO, J., RISSANEN, J.: Agile software development in theory and practice. Universidade de Jyväskylä, Finlândia (Dissertação de Mestrado). 2002.

KEN SCHWABER. "Agile Software Development with Scrum". Microsoft Press, 2004.

LEAN INSTITUTE BRASIL "Desenvolvimento Lean de produtos". 2010. Disponível em: <http://www.lean.org.br> Acesso em: 23 de agosto 2019.

MOLLAHOSEINI Ardakani, MOHAMMAD Reza, Seyyed Mohsen Hashemi, and Mohammadreza Razzazi. "Adapting the scrum methodology for establishing the dynamic inter-organizational collaboration." *Journal of Organizational Change Management* 31.4. 2018.

NASCIMENTO, Marcelino Santos, RODRIGUES, Joseane Pereira. "Scrum aplicado ao desenvolvimento do software em disciplinas de desenvolvimento WEB." *Revista Diálogos Acadêmicos* 6.1. 2018.

PRESSMAN, ROGER S. – Engenharia de Software – Uma Abordagem Profissional – 7º Edição 2011.

RAFAEL SABBAGH "Scrum gestão ágil para projetos de sucesso".2017. Disponível em: [metodoagil.com](http://www.metodoagil.com): <http://www.metodoagil.com.br> Acesso em: 22 de agosto de 2019.

RIBEIRO, Tayse Virgulino, and Cristina D'Ornelas Filipakis Souza. "SIDD–Scrum Iteration Driven Development: Processo Ágil Para Desenvolvimento E Gerenciamento De Software." *Anais Estendidos do XV Simpósio Brasileiro de Sistemas de Informação*. SBC, 2019.

SCHWABER, K. "Scrum Development Process", OOPSLA'95 Workshop on Business Object Design and Implementation. Springer-Verlag (1995).

SCHWABER, K. e Beedle, M. "Agile Software Development with SCRUM", Prentice-Hall, (2002). Disponível em: <http://progettata.com/blog/voce-sabe-as-diferencas-entre-as-metodologias-lean-agile-e-scrum/> Acesso em: 22 de agosto de 2019.

SINGH, Mona. "U-SCRUM: An agile methodology for promoting usability." *Agile 2008 Conference*. IEEE, 2008.

VIEIRA, Denisson, 2014 Disponível em <https://www.mindmaster.com.br> Acesso em: 04 de setembro 2019.

VOGELZANG, Johannes, Wilfried F. Admiraal, and Jan H. van Driel. "Scrum methodology as an effective scaffold to promote students' learning and motivation in context-based secondary chemistry education." *EURASIA Journal of Mathematics, Science and Technology Education* 15. 2019.