



DESENVOLVIMENTO ÁGIL DE SOFTWARE

AGILE SOFTWARE DEVELOPMENT

DESARROLLO ÁGIL DE SOFTWARE

Ana Carolina Andrei Silva¹, Fernando Graciano²

e3122157

<https://doi.org/10.47820/recima21.v3i12.2157>

PUBLICADO: 12/2022

RESUMO

Ágil é uma abordagem interativa para o desenvolvimento de *software* que ajuda as equipes a entregar o produto mínimo viável, promover uma melhor colaboração e lançar *software* de maior qualidade. O objetivo do trabalho foi explorar como esta abordagem tem auxiliado as empresas no mercado atual, assim como tem descrito a literatura de desenvolvimento tecnológico da informação frente às necessidades do mundo globalizado. Trata-se de uma revisão da literatura com pesquisa em livros e artigos científicos das bases de dados da Scielo e Google Books. O resultado mostra ganhos relevantes para a competitividade das empresas, com o uso da metodologia Ágil, que adiciona valor ao negócio, flexibilidade, gestão de tempo, liderança remota, agilidade, geração de mapa de fluxo de valor, gestão inteligente, comunicação eficaz, transparência e engajamento dos colaboradores, eliminação da necessidade de adaptações frequentes, entre outros. Conclui-se que os benefícios da metodologia Ágil estão na sua capacidade de ajudar as equipes em um cenário em evolução, mantendo o foco na entrega eficiente de valor comercial. A cultura colaborativa facilitada pelo Ágil também melhora a eficiência em toda a organização, pois as equipes trabalham juntas e entendem suas funções específicas no processo.

PALAVRAS-CHAVE: *Software. Desenvolvimento Ágil. Benefícios. Competitividade.*

ABSTRACT

Agile is an interactive approach to software development that helps teams deliver the minimum viable product, promote better collaboration, and launch higher quality software. The objective of this work was to explore how this approach has helped companies in the current market, as well as the literature of technological development of information in view of the needs of the globalized world. This is a review of literary research in books and scientific articles from the databases of Scielo and Google Books. The result shows gains relevant to the competitiveness of companies, with the use of agile methodology, which adds value to the business, flexibility, time management, remote leadership, agility, generation of value flow map, intelligent management, effective communication, transparency and employee engagement, elimination of the need for frequent adaptations, among others. It is concluded that the benefits of agile methodology are in its ability to help teams in an evolving scenario, keeping the focus on efficient delivery of business value. The collaborative culture facilitated by Agile also improves efficiency throughout the organization, as teams work together and understand their specific roles in the process.

KEYWORDS: *Software. Agile development. Benefits. Competitiveness.*

RESUMEN

Agile es un enfoque interactivo para el desarrollo de software que ayuda a los equipos a entregar el producto mínimo viable, promover una mejor colaboración y lanzar software de mayor calidad. El objetivo de este trabajo fue explorar cómo este enfoque ha ayudado a las empresas en el mercado actual, así como la literatura de desarrollo tecnológico de la información en vista de las necesidades del mundo globalizado. Esta es una revisión de la investigación literaria en libros y artículos científicos de las bases de datos de Scielo y Google Books. El resultado muestra ganancias relevantes para la competitividad de las empresas, con el uso de metodología Ágil, que agrega valor

¹ Fatec Taquaritinga

² Fatec Taquaritinga



RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

DESENVOLVIMENTO ÁGIL DE SOFTWARE
Ana Carolina Andrei Silva, Fernando Graciano

al negocio, flexibilidad, gestión del tiempo, liderazgo remoto, agilidad, generación de mapa de flujo de valor, gestión inteligente, comunicación efectiva, transparencia y compromiso de los empleados, eliminación de la necesidad de adaptaciones frecuentes, entre otros. Se concluye que los beneficios de la metodología Ágil están en su capacidad para ayudar a los equipos en un escenario en evolución, manteniendo el enfoque en la entrega eficiente de valor comercial. La cultura colaborativa facilitada por Agile también mejora la eficiencia en toda la organización, ya que los equipos trabajan juntos y entienden sus roles específicos en el proceso.

PALABRAS CLAVE: Software. Desarrollo Ágil. Beneficios. Competitividad.

1 INTRODUÇÃO

Como hoje a tecnologia não é mais um *commodity*, mas faz parte do negócio, contém informações e tem grande aplicação do sistema de informação nas empresas, é importante que o gestor saiba tudo o que acontece na organização, para que possa criar novos projetos e investir no seu negócio e em um caminho mais planejado. Assim, em análise e desenvolvimento de sistemas tem uma posição de extrema importância, pois o parceiro da empresa, assume a responsabilidade pelo gerenciamento dos dados e mantém relatórios claros e precisos que auxiliam no planejamento, prevenção e investimento.

O objetivo deste trabalho é incentivar o uso e ressaltar como o desenvolvimento Ágil de *software* pode influenciar no processo de criação de novos projetos e mostrar como ele pode adicionar valor a empresa e quais os benefícios podem trazer diante de sua utilização.

O trabalho se justifica na medida em que se observa que o desenvolvimento Ágil de *software* - também conhecido simplesmente como *Agile* - se trata de uma metodologia de desenvolvimento de *software* que torna possível a aplicação de flexibilidade ao decorrer do processo de desenvolvimento, diferentemente do modelo WATERFALL que apresenta fases específicas que precisam ser devidamente finalizadas para que a próxima etapa se inicie, se tornando um modelo engessado.

Para que o modelo Ágil seja implementado, torna-se necessária uma mudança cultural dentro da organização, onde se difere entre o método tradicional de desenvolvimento de *software* e o Ágil é a adaptabilidade do projeto, ou seja, na metodologia tradicional, mudanças não são bem vistas, pois assim que uma etapa foi inteiramente concluída a opção de retroceder ou alterar algo implica em custos elevados, tornando os projetos cada vez mais engessados e é justamente onde nasce a vantagem da utilização da metodologia Ágil, pois esta disponibiliza que mudanças sejam feitas após o planejamento inicial com ciclos de desenvolvimento menores, tornando também mais fácil a priorização de uma atividade.

Para responder à questão da pesquisa o objetivo foi explorar como está descrito na literatura o desenvolvimento tecnológico da informação frente às necessidades do mundo globalizado.

2 DESENVOLVIMENTO ÁGIL DE SOFTWARE

A Engenharia de *Software* surgiu com o intuito de pesquisar novas ferramentas, técnicas e metodologias, visando um produto de melhor qualidade, um *software* sem falha, buscando atender à



RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

DESENVOLVIMENTO ÁGIL DE SOFTWARE
Ana Carolina Andrei Silva, Fernando Graciano

crescente demanda por *software* e resolver a maioria dos problemas enfrentados pelos desenvolvedores e pelas empresas de desenvolvimento, como o atraso da entrega no determinado prazo, os problemas gerados nos *softwares* por falta de testes e contramedidas (MAX; PRESSMAN, 2016).

Um novo conceito de Engenharia de *Software* surgiu em 2001. Kent Beck e 16 outros desenvolvedores assinaram o “Manifesto para o Desenvolvimento Ágil de *Software*”, também conhecido apenas por “Manifesto Ágil”, sendo uma declaração de princípios e valores que fundamentam o desenvolvimento Ágil de *software*. Ele possui 4 valores e 12 princípios (MAX; PRESSMAN, 2016).

Os quatro valores incluem: indivíduos e interações mais que processos e ferramentas; *software* em funcionamento mais que documentação abrangente; colaboração com o cliente mais que negociação de contratos; responder a mudanças mais que seguir um plano (SOMMWERVILLE, 2015).

Os princípios são: satisfação do cliente atendendo aos requisitos; projetos dinâmicos, com mudanças de rumo no caminho; prazos adequados às necessidades dos clientes; equipes de projeto com multifuncionalidade; motivação de equipe e liderança eficaz; sistema de comunicação horizontal; funcionabilidade já no desenvolvimento do projeto; desenvolvimento sustentável e com participação dos *stakeholders*; excelência técnica e de design; produto enxuto (sem desperdícios); equipes auto-organizáveis; refinamentos e busca de simplificação (MORAES, 2015).

Deve haver iteração do processo de desenvolvimento em que as revisões do sistema são projetadas, implementadas e testadas. Uma diferença crítica é que o primeiro estágio da implementação da mudança pode envolver o entendimento do programa, especialmente se os desenvolvedores do sistema original não forem responsáveis pela implementação da mudança. Durante a fase de compreensão do programa, se deve entender como o programa está estruturado, como ele fornece funcionalidade e como a mudança proposta pode afetar o programa (GOMES, 2013).

Mudanças urgentes podem ter que ser implementadas sem passar por todos os estágios do processo de engenharia de *software*. Se uma falha grave do sistema precisar ser reparada para permitir a operação normal. Se as alterações no ambiente do sistema (por exemplo, uma atualização do sistema operacional) tiverem efeitos inesperados. Se houver mudanças de negócios que exijam uma resposta muito rápida (por exemplo, o lançamento de um produto concorrente) (SOMMWERVILLE, 2015).

A dinâmica de evolução do programa é o estudo dos processos de mudança do sistema. Depois de vários estudos empíricos importantes, Lehman e Belady propuseram que havia uma série de “leis” que se aplicavam a todos os sistemas conforme eles evoluíam. Existem observações sensatas em vez de leis. Eles são aplicáveis a grandes sistemas desenvolvidos por grandes organizações. Não está claro se são aplicáveis a outros tipos de sistema de *software* (MORAES, 2015).



RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

DESENVOLVIMENTO ÁGIL DE SOFTWARE
Ana Carolina Andrei Silva, Fernando Graciano

Com os estudos na área da Engenharia de *Software*, muitos modelos de desenvolvimento diferentes surgiram, como por exemplo o Modelo em Cascata, Modelo Incremental, Modelo de Prototipagem e Modelo Espiral. Esses modelos são conhecidos também como Modelos de Ciclo de Vida (MORAES, 2015).

O objetivo de um mapa de fluxo de valor é visualizar todo o fluxo de um processo do início ao fim e criar um plano para otimizar os esforços para levar a empresa aos resultados desejados. Ao ilustrar o estado atual, um mapa do fluxo de valor permite que se saiba onde existem lacunas atuais (RIGBY; ELK, 2020).

As metodologias ágeis exigem um fluxo contínuo de informações entre os departamentos e todas as partes interessadas. Isso requer uma estrutura de relatórios sólida que requer intervenção manual mínima e pode acomodar cenários únicos (RIGBY; ELK, 2020).

Existem diversas metodologias ágeis, como Scrum, Lean, Kanban ou Smart, cada uma com seu objetivo e características e uso na contabilidade. No caso da e-brom os três métodos usados são a agile, lean e waterfall, sendo as metodologias do aplicativo.

A aplicação metodologia Ágil Scrum é para iniciativas complexas com soluções desconhecidas e alta dependência de *feedback* dos usuários finais. Isso pode ser encontrado no gerenciamento de projetos ao implementar novas tecnologias digitais, aprimorar sistemas existentes, modificar processos para atender a novos requisitos regulamentares. Além disso, à medida que a função a fortalecer sua capacidade analítica, a entrega do scrum pode ser o modo de entrega perfeito para responder com eficiência a solicitações não planejadas de *insights* que informam a tomada de decisões (RICHARD *et al.*, 2018).

O gerenciamento *lean* é uma abordagem de liderança que apoia a melhoria contínua por meio da resolução colaborativa de problemas. Em vez de liderar de cima para baixo, os gerentes lean se esforçam para envolver ativamente suas equipes no processo de melhoria a fim de promover o sucesso pessoal e organizacional. A responsabilidade diária em toda a organização, especialmente nos níveis de liderança, é a chave para a transformação sustentada. Existem quatro razões pelas quais a responsabilidade é tão importante para o gerenciamento lean: confiança, desempenho, domínio, garantia (ARBULO-LOPEZ; FORTUNY-SANTOS, 2010).

Kanban é uma abordagem Ágil que se concentra em melhorar o fluxo de trabalho por meio de um sistema. Também deixa muito espaço etc. Basicamente, Scrum e Kanban são duas ferramentas intimamente relacionadas que ajudam as organizações a se tornarem mais eficazes. As práticas centrais podem ser atribuídas aos valores Kanban de transparência, equilíbrio, cooperação, foco no cliente e fluxo de trabalho. As principais práticas visualizar, tornar as políticas explícitas e complementar *loops* de *feedback* estão relacionadas à Transparência (FULLERTON; KENNEDY; WIDENER, 2013).

Ao criar metas específicas, a melhor abordagem é seguir o modelo *Smart Goals*, um método amplamente usado para tornar o estabelecimento de metas mais eficaz. O acrônimo SMART



RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

DESENVOLVIMENTO ÁGIL DE SOFTWARE
Ana Carolina Andrei Silva, Fernando Graciano

significa: específico, mensurável, acionável, relevante e limitado no tempo (ARBULO-LOPEZ; FORTUNY-SANTOS, 2010).

Com os estudos crescentes na área de engenharia de *software*, logo surgiu o modelo cascata (*waterfall*), um dos modelos mais antigos e conhecidos. Também chamado de “ciclo de vida clássico” e “modelo preditivo”, esse modelo apresenta uma linha do tempo de projeto burocrática e limitada (MOURA; WILDT, 2015).

Mudanças nas forças ambientais fornecem uma oportunidade para as atividades de auditoria mudarem de uma abordagem de planejamento de auditoria reativa e inicial para uma mais ágil. O método Agile, ou Ágil, envolve uma adaptação mais rápida aos ambientes externos e internos e pode resultar em melhor desempenho operacional, qualidade e satisfação do cliente (RICHARD *et al.*, 2018).

Para Richard *et al.*, (2018) a criação de um *software* Ágil foi devido às organizações, que têm grandes investimentos em seus sistemas de *software* - eles são ativos de negócios críticos. Para manter o valor desses ativos para o negócio, eles devem ser alterados e atualizados. A maior parte do orçamento de *software* de empresas é dedicada à mudança e evolução do *software* existente, em vez de desenvolver um novo *software*.

Diante dessa situação, tornou-se nítida a necessidade de estudos para a criação de metodologias de desenvolvimento que apliquem padronização de códigos, gestão de prazo, administração de recursos, planejamento de custo, análise de requisitos, documentação de procedimentos, garantia de qualidade e testes (MOURA; WILDT, 2015).

Um programa que é usado em um ambiente do mundo real deve necessariamente mudar, ou então se tornar progressivamente menos útil naquele ambiente. À medida que um programa em evolução muda, sua estrutura tende a se tornar mais complexa. Recursos extras devem ser dedicados à preservação e simplificação da estrutura (SOMMERVILLE, 2015).

Os modelos evolucionários trabalham com ciclos de desenvolvimento e cada ciclo de desenvolvimento tende a trabalhar um incremento do *software*, enquanto nos modelos evolucionários, o próprio *software* é retrabalhado e melhorado continuamente. Ou seja, o desenvolvimento trabalha naquilo que já foi desenvolvido uma vez para que seja aperfeiçoado (SOMMERVILLE, 2011).

Nos sistemas ágeis de *software* a prototipação consiste em buscar, através de protótipos, um *software* que atenda às necessidades do cliente, em um cenário em que os requisitos iniciais não são suficientes para que o projeto do *software* alcance o objetivo do cliente no primeiro ciclo de desenvolvimento (MAX; PRESSMAN, 2016).

A cada entrega, mais requisitos são descobertos, e assim os objetivos vão ficando cada vez mais claros. Esse conceito pode ser usado em qualquer outro modelo, como uma técnica de aperfeiçoamento contínuo do *software* que está sendo desenvolvido (MAX; PRESSMAN, 2016).

O projeto Ágil leva à construção de um protótipo, que é entregue e avaliado pelos envolvidos, os quais fornecem feedback que é usado para refinar ainda mais os requisitos. A iteração ocorre



RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

DESENVOLVIMENTO ÁGIL DE SOFTWARE
Ana Carolina Andrei Silva, Fernando Graciano

conforme se ajusta o protótipo às necessidades de vários envolvidos e, ao mesmo tempo, possibilita a melhor compreensão das necessidades que devem ser atendidas (MAX; PRESSMAN, 2016).

O modelo espiral também segue o conceito iterativo do modelo evolucionário. Max e Pressman (2016, p.47) o definem como um modelo que “une a natureza iterativa da prototipação aos aspectos sistemáticos e controlados do modelo cascata. Tem potencial para o rápido desenvolvimento de versões cada vez mais completas do *software*”.

Aqui, o processo de *software* é representado como uma espiral, e não como uma sequência de atividades com alguns retornos de uma para outra. cada volta na espiral representa uma fase do processo de *software*, dessa forma, a volta mais interna pode preocupar-se com a viabilidade do sistema; o ciclo seguinte, com definição de requisitos; o seguinte, com o projeto do sistema, e assim por diante.

O *software* Scrum é uma estrutura extremamente popular em desenvolvimento e teste que ajuda a agregar valor ao negócio máximo durante o menor período. Seguindo os 6 princípios Scrum (controle sobre o processo empírico, auto-organização, colaboração, priorização baseada no valor, programação tempo-atividade e iteração) -principais as equipes se tornarão facilmente mais eficientes e coordenadas, e o processo de funcionamento de *softwares* trará melhores resultados com mais rapidez (CARVALHO; MELLO, 2015).

Para Mann e Maurer (2005, p. 70-79):

Apesar de ser uma abordagem relativamente nova, a utilização do método Scrum tem aumentado nos últimos anos, impulsionado pelas recentes pesquisas que mostram que seu uso aumenta a satisfação dos clientes e diminui o atraso em projetos em relação aos métodos tradicionais.

O objetivo geral da gestão de dados é reunir, analisar e comunicar o desempenho de uma empresa - todos os tipos de informações usadas para tomar decisões sobre como administrar bem um negócio, por uma empresa que adota o pensamento enxuto: com foco no valor entregue ao cliente e na eliminação de desperdícios por meio de um melhor fluxo de trabalho e gerenciamento de dados (RUBIN, 2017).

Um sistema Kanban pode ajudar a rastrear o trabalho que está sendo feito, fornecendo uma representação visual do que foi feito, do que está sendo trabalhado e do que ainda precisa ser feito. Também pode ajudar na distribuição do trabalho. Em um quadro Kanban, o trabalho que precisa ser feito é mantido em uma coluna de backlog e pode ser priorizado. Fluxogramas de auditoria também podem ser adicionadas para categorizar o trabalho (por exemplo, retornos individuais vs. empresas) em um único quadro (RUBIN, 2017).

Quando um especialista em gestão de dados termina o que está fazendo, ele pode ir para a coluna do *backlog* e puxar a próxima tarefa de maior prioridade que se encaixa em seu conjunto de habilidades. Kanban também pode ajudar uma empresa a rastrear os recursos. Mais importante ainda, Kanban facilitará a melhoria contínua do processo para que as empresas possam dar aos seus clientes uma experiência incrível que os fará voltar para mais (RUBIN, 2017).



3 PROCEDIMENTOS METODOLÓGICOS

Trata-se de uma pesquisa bibliográfica feita a partir de uma coleta de dados obtidos em livros, artigos científicos e revistas científicas das bases de dados da Scielo e Google Books.

Este é um método de pesquisa no qual todos os assuntos listados possuem uma problemática central em relação as empresas com o desenvolvimento Ágil de *software*.

Foram respeitados os direitos autorais das literaturas utilizadas neste estudo, conforme determinado na Lei 9610 de 19 de fevereiro de 1998.

4 RESULTADOS E DISCUSSÃO

O Quadro 1 mostra problemas comuns do uso das planilhas Excel numa empresa, com base nos autores pesquisados na revisão da literatura

Erros humanos – inserção manual de dados – erro de digitação – deleção de dados relevantes
Versões conflitantes – uso de versão atualizada e antiga
Dificuldade no acesso às informações – arquivos em mais de um computador
Validação de dados – a planilha pode aceitar erros ou nomenclaturas
Tempo perdido na navegação – planilhas com muitas linhas, guias, planilhas vinculadas
Falhas de segurança – funcionário não seguir medidas de segurança e a criptografia não é forte
Lentidão na tomada de decisão – consolidar e extrair dados de computadores diferentes e problema de integridade de dados
Não há produtividade – baixo rendimento e eficácia
Muitas atividades repetitivas
Não há hospedagem de dados na nuvem – sem acesso por dispositivo com segurança
Não há gerenciamento em tempo real – não há sistema online integrado
Não há módulos interligados – não há repasse de dados de forma inteligente

Quadro 1 – Problemas com planilhas Excel.
Fonte: A autora do trabalho (2022)

Haverá um avanço notável na tecnologia com a chegada da computação em nuvem que acaba tornando essencial o entendimento e até mesmo as exigências dos clientes. A computação em nuvem permitirá o armazenamento e acessibilidade de dados *on-line* - na nuvem - em vez de em um disco rígido. O acesso a programas pela nuvem permitirá um fluxo livre de informações, independentemente de onde o funcionário esteja ou do dispositivo que estiver usando. A nuvem também oferecerá aos clientes e colaboradores da empresa a capacidade de acessar certos dados, facilitando a iteração e a troca de informações. A computação em nuvem será um grande negócio na empresa, e poderá se estabelecer a empresa como um líder com visão de futuro na área, atualizando seu conhecimento das opções populares de *software* de contabilidade disponíveis apenas na nuvem. Isso inclui QuickBooks Online, Kashoo, Xero e FreshBooks (RIGBY; ELK, 2020).



RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

DESENVOLVIMENTO ÁGIL DE SOFTWARE
Ana Carolina Andrei Silva, Fernando Graciano

O Quadro 2 mostra as vantagens do uso do Ágil numa empresa, com base nos autores pesquisados na revisão da literatura.

A utilização do Ágil para otimizar <i>software</i> para o escritório da empresa possibilita reduzir de forma bastante sensível os custos com mão de obra e outras despesas operacionais, menor espaço
Eliminação de atividades repetitivas
As atividades serão mais seguras, bem como <i>backup</i> , uso de <i>login</i> e senha
Tarefas são feitas automaticamente, dentro de um processo que possibilita reduzir o índice de erros humanos
Todas as tarefas se tornam integradas, permitindo que o sistema possa gerar relatórios com maior rapidez e precisão
Somente um arquivo central, com mínima possibilidade de erros
Automaticamente, a otimização do processo vai gerar maior qualidade nos resultados apresentados para os clientes, oferecendo maior satisfação, ao mesmo tempo em que o escritório é mais reconhecido pela sua eficiência
Saber o que é prioritário para os clientes e para o escritório
Sistema contábil que ofereça armazenamento em nuvem ilimitado, evitando que o escritório tenha maiores custos no futuro e, ao mesmo tempo, contando com uma tecnologia mais segura para o armazenamento das informações
Estrutura inteligente de pastas para facilitar a organização de maior volume de dados e informações
Possibilidade de compartilhamento interno, com rastreabilidade, em tempo real, sem depender de outras formas de comunicação
Maior rapidez e agilidade nos processos e maior autonomia para os usuários
Interface simples, moderna e intuitiva
Integração com outras interfaces, principalmente aquelas implantadas pelos órgãos governamentais
Versão para demonstração por um prazo determinado
Suporte para atender às necessidades do escritório, podendo atender a qualquer momento
Oferecer mais aos clientes, garantindo um trabalho eficiente e com a qualidade por eles exigida
Tecnologia automatizada

Quadro 2 – Ganhos com a adoção do Ágil.
Fonte: A autora do trabalho (2022)

Ganhos relevantes para a competitividade de uma empresa com o uso do Ágil, segundo Rubin (2017).

- a) Adiciona valor ao negócio;
- b) Flexibilidade;
- c) Gestão de tempo;
- d) Liderança remota;
- e) Agilidade;
- f) Geração de mapa de fluxo de valor;
- g) Gestão inteligente – controle detalhado de tarefas;



- h) Comunicação ideal;
- i) Transparência e engajamento dos colaboradores;
- j) Eliminação da necessidade de adaptações frequentes.

O mapeamento do fluxo de valor (VSM) revela informações sobre as etapas do processo que uma empresa usa para criar valor. É usado para agilizar os fluxos do processo, reduzindo assim as atividades sem valor agregado. Dependendo do formato utilizado, pode apontar informações como: o tempo de trabalho e o tempo de espera necessários para cada etapa de um processo; a quantidade de mão de obra necessária para uma etapa de trabalho, incluindo a identificação de horas extras; a taxa de erro por etapa de trabalho; tempo de inatividade por etapa de trabalho. Os mapas do fluxo de valor podem ser usados para revelar o estado atual de um processo ou para esclarecer seu estado futuro, uma vez que uma série de melhorias tenham sido implementadas. Também pode ser aplicado aos diversos processos contábeis. Os gráficos resultantes podem ser usados para apontar áreas que precisam de melhorias, como redução de erros, automação para eliminar o tempo da equipe e controles alterados para encurtar os fluxos de processo (MORAES, 2015; RIGBY; ELK, 2020; RUBIN, 2017).

A multitarefa diminui a produtividade. Muitos estudos provaram que quanto mais pessoas trocam de tarefas, menos eficazes e mais propensas a erros elas são ao concluir o trabalho. Kanban ajuda a eliminar essas penalidades de mudança de contexto, limitando o trabalho em andamento. O trabalho é puxado pelo sistema pela capacidade disponível, não empurrado com base no fornecimento. Em outras palavras, as pessoas terminam o que estão fazendo antes de começar algo novo. O aumento da eficiência acabará por reduzir o tempo que leva para um item percorrer o sistema Kanban (RUBIN, 2017).

5 CONCLUSÃO

O trabalho possibilitou apresentar os benefícios que o desenvolvimento Ágil de *software* traz para à equipe, e neste aspecto foram exploradas as metodologias ágeis, considerando mapa de fluxo de valor, kanban e sistemas Ágeis.

Com o Ágil se tem uma visão macro de todos os fluxos de valor e controle, possibilitando visualizar em detalhes todas as atividades e tarefas. Um fator relevante é a comunicação na empresa, para que haja interação com os *stakeholders*. Se trabalha estrategicamente, com transparência, *lead time* reduzido, combate ao desperdício, sinergia dos colaboradores

Os resultados mostram que ganhos relevantes para a competitividade de uma empresa com o uso do Ágil, pois adiciona valor ao negócio; flexibilidade; gestão de tempo; liderança remota; agilidade; geração de mapa de fluxo de valor; gestão inteligente – controle detalhado de tarefas pelo kanban; comunicação eficaz; transparência e engajamento dos colaboradores; eliminação da necessidade de adaptações frequentes, entre outros.



RECIMA21 - REVISTA CIENTÍFICA MULTIDISCIPLINAR ISSN 2675-6218

DESENVOLVIMENTO ÁGIL DE SOFTWARE
Ana Carolina Andrei Silva, Fernando Graciano

Conclui-se que para o profissional da análise e desenvolvimento de sistemas o Ágil serve para resolver problemas até então sem experiência em muitas empresas e cria soluções que valorizam a empresa diante da competitividade global e busca da produtividade. Ágil foi desenvolvido para tornar as empresas mais ágeis, por meio de processo digital e mapeamento de fluxo de valor.

REFERÊNCIAS

ARBULO-LOPEZ, P. R. DE; FORTUNY-SANTOS, J. An accounting system to support process improvements: Transition to lean accounting. **Journal of Industrial Engineering and Management**, v. 3, n. 3, p. 576–602, 2010.

CARVALHO, B. V.; MELLO, C. H. P. Aplicação do método ágil scrum no desenvolvimento de produtos de software em uma pequena empresa de base tecnológica. **Gest. Prod.**; v. 19, n. 3, p. 1-10, 2015.

FULLERTON, R. R.; KENNEDY, F. A.; WIDENER, S. K. Management accounting and control practices in a lean manufacturing environment. **Accounting, Organizations and Society**, v. 38, n. 1, p. 50-71, 2013.

GOMES, A. F. **Agile Desenvolvimento de software com entregas frequentes e foco no valor do negócio**. São Paulo: Casa do Código, 2013.

MARCONI, M. A.; LAKATOS, E. M. **Fundamentos de Metodologia Científica**. São Paulo: Atlas, 2017.

MAX, B. R.; PRESSMAN, R. S. **Engenharia de Software uma abordagem profissional**. 8. Ed. São Paulo: AMGH, 2016.

MORAES, M. V. G. **Sistema de Gestão Princípios e Ferramentas**. São Paulo: Erica, 2015.

MOURA, D.; WILDT, D. **Extreme Programming: práticas para o dia a dia no desenvolvimento ágil de software**. São Paulo: Casa do Código, 2015.

RICHARD, I. *et al.* Agility in Audit: Could Scrum Improve the Audit Process? **Current Issues in Auditing**; v. 12, n. 1, p. 18-28, 2018.

RIGBY, D.; ELK, S. **Ágil do Jeito Certo**: Transformação sem caos. Rio de Janeiro: Benvirá, 2020.

RUBIN, K. S. **Scrum essencial**: um guia prático para o mais popular processo ágil. São Paulo: Alta Book, 2017.

SOMMERVILLE, I. **Engenharia de Software**, São Paulo: Pearson Education, 2011.