



FM2S.COM.BR
EAD.FM2S.COM.BR

METODOLOGIAS ÁGEIS



METODOLOGIAS ÁGEIS

AUTORES

Dr. Virgílio F. M. dos Santos

Master Black Belt, Professor Extensão
Unicamp (IMECC e Economia), Engenheiro
Mecânico, Mestre e Doutor na área.

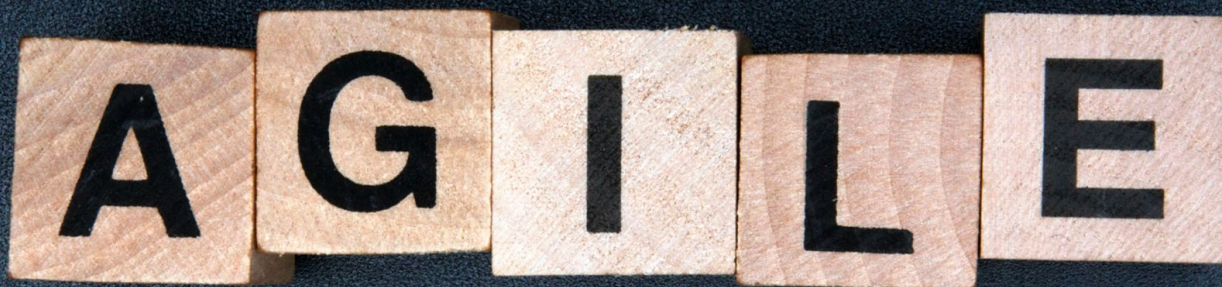
Dr. Murilo F. M. dos Santos

Master Black Belt, Engenheiro Mecânico,
Mestre e Doutor na área.

SUMÁRIO



Introdução	4
O que são Metodologias Ágeis?	5
Manifesto Ágil?	8
Qual a origem do Manifesto Ágil?	8
Os princípios do Manifesto Ágil	9
Quais são as Metodologias Ágeis?	16
Scrum	16
Kanban	19
Programação Extrema (XP)	22
Benefícios da Metodologia Ágil	24



INTRODUÇÃO

Metodologias ágeis são diversas metodologias para a gestão de equipes multifuncionais, que focam em melhorar a eficiência da comunicação e trabalhos de times, de maneira simples e desburocratizada. Começaram sendo aplicadas para o desenvolvimento de softwares (como é o caso do **SCRUM**), mas evoluíram para outras atividades, de desenvolvimento ou de gestão.

As metodologias ficaram famosas com o **manifesto ágil**, uma declaração dos princípios que norteiam a maioria das metodologias. No entanto, nos últimos anos, uma nova maneira de criar software levou o seu desenvolvimento e o mundo dos testes à tona: Agile.

De fato, de acordo com o State of Agile Report da VersionOne, a partir de 2017, 94% das organizações praticam o Agile de alguma forma. Entretanto, os entrevistados relatam que essa adoção nem sempre é difundida em suas empresas, o que significa que ainda há um longo caminho a percorrer em termos de adoção e maturidade.



O QUE SÃO METODOLOGIAS ÁGEIS?

O surgimento da metodologia como um enorme movimento global, que vai além do software, é impulsionado pela descoberta de que a única maneira de as organizações lidarem com o turbulento mercado atual, orientado para o cliente, é tornar-se ágil. Essa metodologia permite que **as organizações dominem a mudança contínua** e que as empresas floresçam em um mundo cada vez mais volátil, incerto, complexo e ambíguo.

Ao abandonar o risco de passar meses ou anos em um processo que acaba por falhar devido a algum pequeno erro em uma fase inicial, confia em funcionários e equipes que trabalham diretamente com os clientes para entender as metas e fornecer soluções de maneira rápida e incremental.

O que torna o Agile difícil de ser compreendido por alguns gerentes é que não é apenas uma metodologia ou processo que pode ser implementado dentro das suposições atuais de uma empresa. **A maneira Ágil de administrar uma organização redefine o próprio conceito de uma corporação que prevaleceu nos últimos cem anos.**

Em vez de uma corporação ser concebida como uma eficiente máquina estável destinada a explorar seu modelo de negócios existente, a organização Ágil é um organismo vivo em crescimento, adaptando e aprendendo que está em constante fluxo para explorar novas oportunidades e agregar novos valores aos clientes.

A **Metodologia Ágil** é uma abordagem focada em pessoas e também em resultados para o **desenvolvimento de software que respeita o mundo em rápida transformação**. Além disso, centra-se em torno de planejamento adaptativo, auto-organização e prazos de entrega curtos. É flexível, rápida e visa melhorias contínuas na **qualidade**, usando ferramentas como Scrum e Extreme Programming.

Essa metodologia faz o indivíduo perceber que o velho método “cascata” de desenvolvimento de software deixa muito a desejar, ou seja, que processo de “planejar, projetar, construir, testar, entregar” funciona bem para fazer carros ou edifícios, mas não tão bem para criar sistemas de software. Em um ambiente de **negócios** no qual o hardware, a demanda e a concorrência são variáveis e que mudam rapidamente, o ágil funciona percorrendo a linha tênue entre muitos processos e não o suficiente.

A metodologia ágil conquistou o mundo do desenvolvimento de software e rapidamente cimentou seu lugar como “o padrão ouro”. Essas metodologias estão enraizadas no planejamento adaptativo, na entrega antecipada e na melhoria contínua, tudo com o objetivo de poder responder às mudanças de maneira rápida e fácil. Como resultado, não é surpresa que 88% dos entrevistados no Relatório de Estado Ágil do VersionOne de 2017 classificaram a capacidade de se adaptar à mudança como o benefício número um de adotar o Agile.

O Ágil reconhece que o futuro de uma empresa depende de inspirar aqueles que fazem o trabalho para acelerar a inovação e agregar valor genuíno aos clientes. Reconhece que aumentar a capacidade de quem faz o trabalho depende de dar autonomia a equipes auto-organizadas dentro de parâmetros amplos de controle. **Valoriza a transparência e a melhoria contínua a frente da previsibilidade e da eficiência**. Reconhece que as conversas interativas abertas são mais valiosas do que as diretivas descendentes. Ele para de fazer qualquer coisa que não agregue valor aos clientes finais. Percebe que a chave para o sucesso não é fazer mais trabalho mais rápido, mas sim ser mais inteligente, gerando mais valor com menos trabalho e entregando-o mais cedo.

Quando o Agile é feito corretamente, todas as partes da organização estão continuamente explorando como agregar mais valor aos clientes. Isso não só cria significado para aqueles que realizam o trabalho e encanta aqueles para quem o trabalho é feito: ele resulta em retornos generosos para a própria organização.

Para seus defensores, a metodologia Ágil é uma maneira genuinamente melhor de administrar uma empresa e uma economia – melhor para aqueles que fazem o trabalho, melhor para aqueles para quem o trabalho é feito, melhor para a própria organização. Em vez de a administração extrair valor da empresa, **o Ágil gera valor para os clientes e para a sociedade como um todo.**

Dica: Nesta [apostila](#) totalmente gratuita,, será abordado o SCRUM, um conjunto de práticas para a gestão ágil de projetos. Com esta técnica, as equipes são capazes de produzir mais em menos tempo.



MANIFESTO ÁGIL

Qual a origem do Manifesto Ágil?

O Manifesto Ágil é uma declaração elaborada pelos autores precursores do que hoje conhecemos como Metodologias Ágeis, que abordam **12 princípios** e **4 valores** para o desenvolvimento ágil de softwares. Embora tenha tido seu começo no desenvolvimento de TI, esses princípios do manifesto são aplicados atualmente nos mais diversos ramos da gestão. Dessa forma, mesmo já existindo há um tempo considerável, o teste continua bastante relevante, como comentamos em no post "[O Manifesto Ágil continua relevante?](#)".

No início de 2001, em Snowbird, Utah, 17 pessoas se reuniram para discutir o futuro do desenvolvimento de software. Os membros do grupo compartilhavam uma frustração sobre o estado das coisas, mesmo que discordassem sobre como remediar a situação. **O problema, eles concordavam, era que as empresas estavam tão focadas em planejar e documentar excessivamente seus ciclos de desenvolvimento de software que perdiam de vista o que realmente importava – satisfazer seus clientes.**

As empresas podem ter promovido valores corporativos como "excelência" e "integridade", mas esses valores pouco fizeram para guiar as pessoas – especialmente os desenvolvedores de software – em direção a um caminho melhor. Isso precisava mudar. Muitos dos Snowbird 17 já tinham ideias

sobre como inaugurar a nova era do desenvolvimento de software.

O Manifesto Ágil emergiu deste fim de semana prolongado; com apenas 68 palavras, o documento curto e doce mudou para sempre o desenvolvimento de software. Nas quase duas décadas desde sua criação, essas palavras (e os 12 princípios que se seguem) foram adotadas (em graus variados) por inúmeros indivíduos, equipes e empresas.

Em outras palavras, **os princípios ágeis já existiam antes do Manifesto Ágil**: as pessoas aplicaram-nos ao desenvolvimento de software e então esses valores foram capturados no Manifesto Ágil. Então, as pessoas passaram a seguir os princípios do Manifesto e a aplicá-los em seu próprio trabalho. Aliás, com toda a ideia de reciclagem, é hora de uma atualização do Manifesto Ágil? Não necessariamente.

Quando algo tão culturalmente importante quanto o Manifesto aparece, você até pode reinterpretá-lo, mas não há nada como o original. Então, em vez de tentar atualizá-lo oficialmente, talvez seja melhor descobrir como aplicá-lo a seu caso, em sua equipe ou organização.

Os princípios do Manifesto Ágil

Os princípios do Manifesto Ágil são os mais importantes alicerces de todas as metodologias ágeis:

Princípio Ágil 1 - "Nossa maior prioridade é satisfazer o cliente através da entrega antecipada e contínua de software valioso".

Como fica na prática:

- As equipes de produto usam produtos mínimos viáveis e experimentação rápida para testar hipóteses e validar ideias;
- Versões frequentes ajudam a alimentar um ciclo de feedback contínuo entre cliente e produto;
- Enviado e pronto não são a mesma coisa. Em vez de liberar um produto

“acabado”, as iterações continuam a fazer melhorias incrementais no produto com base no feedback do cliente e do mercado.

Princípio Ágil 2 - “Bem-vindo a mudanças nos requisitos, mesmo no final do desenvolvimento. Os processos ágeis aproveitam a mudança para a vantagem competitiva do cliente.”

Como fica na prática:

- As equipes de produtos são orientadas por objetivos estratégicos de alto nível e talvez até por temas abaixo desses objetivos. O sucesso do departamento de produtos é medido em relação ao progresso em direção a essas metas estratégicas, e não pela entrega de um conjunto de recursos predefinidos;
- O produto está constantemente atento ao solo, monitorando o mercado, o feedback do cliente e outros fatores que podem influenciar a direção do produto. Quando o insight acionável é descoberto, os planos são ajustados para melhor atender às necessidades dos clientes e dos negócios;
- A estratégia do produto e os planos táticos são revisados, ajustados e compartilhados regularmente para refletir mudanças e novas descobertas. Como tal, o produto precisa gerenciar adequadamente as expectativas dos envolvidos executivos e garantir que eles entendam o porquê das mudanças.

Princípio Ágil 3 - “Forneça software de trabalho com frequência, de algumas semanas a alguns meses, com preferência pela menor escala de tempo.”

Como fica na prática:

- Os ciclos de desenvolvimento ágil, geralmente chamados de “sprints” ou “iteraões”, dividem as iniciativas de produtos em partes menores que podem ser concluídas em um prazo definido. Frequentemente, esse perío-

do varia de 2 a 4 semanas, o que é realmente um sprint se você considerar os ciclos de desenvolvimento semelhantes às maratonas que as equipes em cascata costumam seguir;

- Outra alternativa popular aos sprints ágeis é a implantação contínua. Esse método de envio de software geralmente funciona menos em termos de prazos pré-determinados e mais em termos de simplesmente decidir o que fazer e fazê-lo.

Princípio Ágil 4 - “Os empresários e desenvolvedores devem trabalhar juntos diariamente durante todo o projeto.”

Como fica na prática:

- As equipes multifuncionais de desenvolvimento de produtos ágeis incluem as pessoas do produto. Isso significa que o produto é representado na equipe de desenvolvimento, preenchendo a lacuna entre os aspectos técnicos e de negócios do produto;
- Reuniões diárias de atualização (ou stand-ups) são uma técnica que muitas lojas ágeis usam para colocar esse princípio em prática e manter todos conectados.

Princípio Ágil 5 - “Crie projetos em torno de indivíduos motivados. Dê a eles o ambiente e o apoio de que precisam e confie neles para fazer o trabalho.”

Como fica na prática:

- O produto deve garantir claramente que a engenharia entende a estratégia e os requisitos antes do início do desenvolvimento. Isso significa não apenas compartilhar histórias de usuários com a equipe multifuncional, mas também a imagem maior descrita no roteiro do produto.
- O produto não é responsável por explicar “como” algo deve ser construído:

ele precisa compartilhar o que e por quê, cabendo à equipe de entrega determinar como. Além disso, o produto não gerencia o resultado durante os sprints, mas estes se tornam disponíveis para responder perguntas e fornecer suporte, conforme necessário.

Princípio Ágil 6 - “O método mais eficiente e eficaz de transmitir informações para e dentro de uma equipe de desenvolvimento é a conversa cara a cara.”

Como fica na prática:

- Reuniões de stand-up diárias;
- Sessões colaborativas de preparação de pedidos em atraso;
- Reuniões de planejamento da Sprint;
- Demonstrações frequentes;
- Programação em pares.

Princípio Ágil 7 - “O software de trabalho é a principal medida de progresso.”

Como fica na prática:

- Projetar e liberar “Recursos Mínimos Viáveis” em vez de conjuntos de recursos totalmente desenvolvidos significa pensar antes de tudo nas pequenas coisas que podemos enviar para começar a receber feedback e validar os clientes enquanto continuamos a desenvolver o software;
- Uma mentalidade de falha rápida significa avançar mesmo em tempos de incerteza e testar ideias rapidamente;
- Envie o software com frequência: um produto útil agora é melhor do que um produto perfeito mais tarde.

Princípio Ágil 8 - “Os processos ágeis promovem o desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem poder manter um ritmo constante indefinidamente.”

Como fica na prática:

- Antes de cada sprint, é feita uma consideração cuidadosa da quantidade de trabalho que pode ser comprometida. As equipes de desenvolvimento não prometem o que podem e o que não podem entregar. As estimativas de esforço são uma prática comum no estabelecimento de expectativas de produto para as equipes de desenvolvimento;
- Todos concordam com o que será feito durante um sprint. Uma vez iniciado o sprint, nenhuma tarefa adicional deve ser adicionada, exceto em casos raros;
- Os gerentes de produto devem atuar como porteiros para reduzir o ruído de outras partes interessadas e evitar a necessidade de trabalhos adicionais não planejados durante um sprint em andamento;
- O pessoal do produto deve fazer sua parte na promoção de uma sensação de segurança psicológica em toda a equipe multifuncional, que incentive a comunicação aberta e o feedback que flui livremente.

Princípio Ágil 9 - “A atenção contínua à excelência técnica e ao bom design aumenta a agilidade.”

Como fica na prática:

- A equipe precisa estar ciente da dívida técnica e das suas implicações sobre quaisquer novos recursos ou iniciativas adicionados ao backlog. Desenvolvedores e produtos precisam trabalhar juntos para entender se e quando a dívida técnica é aceitável;
- Regularmente, o produto precisará alocar recursos de desenvolvimento para esforços de refatoração. A refatoração não pode ser uma reflexão

tardia, mas uma consideração contínua.

Princípio Ágil 10 - "Simplicidade – a arte de maximizar a quantidade de trabalho que não precisou ser realizado – é essencial".

Como fica na prática:

- Os gerentes de produto precisam tomar decisões de produto muito focadas e alinhar estreitamente a estratégia do produto com as metas organizacionais, ao mesmo tempo em que são extremamente exigentes quanto às histórias e recursos do usuário. O uso de técnicas de priorização de iniciativas por esforço e impacto previsto é uma das maneiras pelas quais as equipes de produtos podem aplicar esse princípio ágil ao desenvolvimento de produtos;
- Os sprints curtos que caracterizam o ágil, por apresentar muitas oportunidades para testes e experimentações rápidos, podem ajudar a reduzir a incerteza sobre se as iniciativas que realmente terão o impacto previsto. Usar experimentos para validar ideias antes de colocá-las em prática de acordo com as especificações é uma ótima maneira de eliminar ideias ruins e identificar boas.

Princípio Ágil 11 - "As melhores arquiteturas, requisitos e projetos emergem das equipes auto-organizadas".

Como fica na prática:

- As equipes auto-organizadas são grupos autônomos dentro da organização que assumem o controle e a responsabilidade sobre seus respectivos projetos e são proprietários dessas áreas. Diferentes organizações praticam esse princípio de maneiras diferentes. O Spotify, por exemplo, usa "esquadrões de produtos" para essa prática.

Princípio Ágil 12 - “A intervalos regulares, a equipe reflete sobre como se tornar mais eficaz, e depois ajusta seu comportamento de acordo.”

Como fica na prática:

- A experimentação e o teste não se limitam apenas ao produto. As equipes ágeis são incentivadas a experimentar seus processos. Dessa forma, você pode pensar que já está fazendo algo bem, apenas para testar uma versão revisada do processo e descobrir um método ainda mais eficaz. Assim, experimentar seu processo e sua equipe é tão importante quanto experimentar o software que você está construindo;
- Retrospectivas regulares são oportunidades para a equipe discutir o que correu bem, o que não correu tão bem e onde o processo pode ser ajustado para melhorar as coisas no futuro. Por conseguinte, elas são um excelente meio para gerentes e proprietários de produtos saberem se estão se comunicando efetivamente com os desenvolvedores e se estão dando-lhes o suporte necessário antes, durante e depois dos sprints;
- Outra consideração a ser feita em relação a esse princípio ágil é que, para praticá-lo efetivamente, você precisa criar uma cultura de confiança e transparência que incentive a abertura e o compartilhamento frequente de feedback.



QUAIS SÃO AS METODOLOGIAS ÁGEIS?

Cada organização é única e enfrenta fatores internos diferentes (ou seja, tamanho da organização e partes interessadas) e fatores externos (por exemplo, clientes e regulamentos). Para ajudar a atender às diversas necessidades de diferentes organizações, existem várias metodologias Ágeis, por isso a combinação certa para a sua equipe dependerá dos fatores, necessidades e objetivos internos e externos. Os exemplos mais populares e comuns são Scrum, Kanban, Extreme Programming (XP), Desenvolvimento Dirigido por Funções (FDD), Método de Desenvolvimento de Sistemas Dinâmicos (DSDM), Desenvolvimento de Software Adaptável (ASD), Crystal e Desenvolvimento de Software Lean (LSD).

SCRUM

O que é?

Uma das mais populares metodologias de teste de software (usada por 58% das organizações que adotaram o Agile, de acordo com o VersionOne), o Scrum adota uma abordagem altamente iterativa que se concentra na definição dos principais recursos e objetivos antes de cada sprint. Ele é projetado para reduzir o risco e, ao mesmo tempo, fornecer valor rapidamente.

O Scrum começa com um requisito ou uma história de usuário que descreve como os recursos devem ser executados e testados. A equipe, em seguida, percorre uma série de sprints para fornecer pequenas explosões de valor rapidamente. Para ajudar a equipe a trabalhar dessa maneira flexível e evitar a mudança de prioridades, o Scrum requer que as perguntas sejam respondidas desde o início.

Como isso é diferente da Waterfall?

Enquanto a esta inclui vários ciclos de teste e correção de bugs antes de lançar um produto, o Scrum é muito mais colaborativo e iterativo. Uma das maiores diferenças é que a Waterfall exige documentação pesada desde o início, o que dificulta a troca de recursos à medida que o processo avança, podendo ser negativo em alguns ambientes (como software para consumidor) e positivo em outros (como aqueles em que a equipe está tentando lançar um foguete, ninguém quer requisitos para algo perigoso mudando frequentemente). Dito isso, você pode pensar em Scrum como muitas "mini-cachoeiras", já que os requisitos são bem definidos no início de cada sprint e não devem mudar dentro dele. A diferença é que os requisitos detalhados para o próximo sprint não são definidos com meses de antecedência.

Mergulhando mais fundo, o Scrum exige mais colaboração regular entre testadores, desenvolvedores e BAs, geralmente na forma de levantamentos diários e retrospectivas de sprint, para garantir a comunicação e o alinhamento adequados. Além disso, existe um Scrum Master que ajuda a manter o projeto na tarefa, removendo os bloqueadores da equipe para garantir que eles sejam mais eficazes. O Scrum Master pode ser qualquer pessoa da equipe, como um desenvolvedor ou um testador.

O que a adoção implica?

O Scrum oferece uma das transições mais fáceis para equipes provenientes de um ambiente do Waterfall, pois é baseado em tempo com

sprints e os lançamentos ainda podem ser planejados com antecedência. Dito isto, exige iterações mais rápidas e colaboração mais forte.

Para quem é recomendado?

Por causa de suas iterações rápidas, o Scrum é mais adequado para equipes cujos clientes e partes interessadas desejam estar ativamente envolvidos, vendo regularmente produtos de trabalho em reuniões de demonstração. Essa colaboração permite que a equipe faça alterações para as próximas vitrines. Os principais membros da equipe que devem estar envolvidos ao adotar uma abordagem Scrum incluem:

- Dono do produto;
- Scrum Master;
- Desenvolvedores;
- Engenheiros de Automação;
- Testadores;
- Stakeholders.

Quais são as melhores práticas?

Além de uma forte comunicação, colaboração e adaptabilidade, outras práticas recomendadas para testadores seguindo uma metodologia Scrum incluem:

- Determinar os critérios de aceitação com base na comunicação (normalmente na forma de uma história do usuário) de um representante de vendas ou cliente;

Nota: essa conexão direta deve ajudar a reduzir as falhas de comunicação.

- Utilizar os critérios de aceitação para desenvolver código e garantir a sua aprovação pela equipe.
- Testar o código em ambientes semelhantes à sandbox, bem como em ambientes semelhantes à produção, antes de implantá-lo em produção.

O Scrum é um sistema prático que consiste em etapas e componentes

simples de intertravamento:

- O proprietário de um produto faz uma lista de desejos priorizada, conhecida como um backlog de produto;
- A **equipe** do Scrum leva uma pequena parte do topo da lista de desejos chamada de sprint backlog e planeja implementá-la;
- A equipe conclui sua tarefa de backlog de sprint em um sprint (um período de 2 a 4 semanas). Eles avaliam o progresso em uma reunião chamada um scrum diário;
- O ScrumMaster mantém a equipe focada no objetivo;
- No final do sprint, o trabalho está pronto para ser enviado ou exibido. A equipe fecha o sprint com uma revisão e, em seguida, inicia um novo sprint.

Aqui está um exemplo de como o Scrum funciona: Bill se encontra com um cliente para discutir as necessidades de sua empresa. Essas necessidades são o backlog do produto. Bill escolhe as tarefas mais importantes para trabalhar nas próximas duas semanas. Sua equipe se reúne diariamente para direcionar o trabalho para o dia seguinte e lidar com os obstáculos. No final do sprint, Bill entrega o trabalho, analisa o backlog e define a meta para o próximo sprint. O ciclo se repete até que o software esteja completo.

Dica: A certificação de Especialista SCRUM visa capacitar o aluno na metodologia SCRUM de maneira rápida e prática. Também existem exercícios para quem está buscando a certificação de SCRUM Master.

KANBAN

O que é?

O Kanban é uma metodologia muito simples baseada no Agile, baseada na fabricação, visto que foi desenvolvida pela Toyota para ajudar a aumentar a produtividade nas fábricas. Em essência, o Kanban pode ser considerado uma grande lista de tarefas prioritárias. Como no Scrum, os requisitos nessa metodolo-

gia são rastreados pelo estágio atual no processo (tarefas, no desenvolvimento, no teste, concluído).

Ao contrário do Scrum, o Kanban não é baseado no tempo. Pelo contrário, baseia-se apenas na prioridade. Quando um desenvolvedor está pronto para a próxima tarefa, ele o puxa da lista de tarefas. Como há menos reuniões de planejamento, essa abordagem significa que a equipe precisa estar extremamente próxima. Nesse tipo de ambiente, se os desenvolvedores trabalharem muito mais rápido que os testadores, os gargalos vão surgir. Nessas situações, qualquer pessoa da equipe deve entrar e ajudar em diferentes áreas. É claro que atender a essa necessidade requer muita flexibilidade e adaptabilidade.

Como isso é diferente da Waterfall?

O Kanban ainda tem requisitos como o Waterfall, mas eles podem mudar, pois a equipe de testes não começa a pensar em testar cada requisito até que o desenvolvedor o selecione na parte superior do backlog. Em contraste, a Waterfall é fortemente baseada no tempo, com muita sobrecarga no planejamento. O planejamento pesado que vem em um ambiente do Waterfall é ótimo em alguns casos, como ao criar itens caros, mas nem sempre é necessário. Com o Kanban, os lançamentos ainda são planejados, mas as equipes geralmente não prometem recursos a determinadas datas, a menos que o item em questão esteja próximo do topo do backlog.

O que a adoção implica?

O Kanban oferece uma transição simples para as equipes certas. Para fazer uma transição suave para o Kanban, os analistas de negócios, desenvolvedores, testadores e partes interessadas devem se sentar juntos e se comunicar regularmente. Ao fazer a transição para o Kanban, é importante lembrar que essa metodologia oferece a maneira mais rápida de levar o código à produção, mas é provável que o código tenha alguma dívida técnica. Isso porque

desenvolver sem sempre saber o que virá a seguir não se presta necessariamente à produção do código mais reutilizável.

Para quem é recomendado?

O Kanban é mais adequado para pequenas equipes ou equipes que não produzem recursos para o público e/ou prometem determinadas datas para lançamentos. Além disso, é uma excelente metodologia de escolha para quaisquer produtos ou equipes focados principalmente no trabalho de manutenção, já que os bugs nem sempre são diretos e geralmente exigem pesquisas para serem resolvidas, o que dificulta o gerenciamento do tempo. Equipes que não podem minimizar a quantidade de planejamento para problemas provavelmente estarão em melhor situação seguindo uma metodologia Scrum ou Waterfall.

Os principais membros da equipe que devem estar envolvidos em um ambiente Kanban incluem:

- Dono do produto;
- Gestor de projeto;
- Desenvolvedores;
- Engenheiros de Automação;
- Testadores;

Quais são as melhores práticas?

Além de manter a visibilidade e priorizar a colaboração, as práticas recomendadas para os testadores que seguem uma metodologia Kanban incluem:

- Manter linhas de comunicação muito abertas entre os proprietários de empresas, desenvolvedores e testadores;
- Garantir que a equipe tenha flexibilidade para assumir outras funções fora de suas principais responsabilidades, a fim de ajudar a eliminar gargalos;
- Fazer de todos um proprietário do produto para que eles se importem

com o resultado.

O Método Kanban é usado pelas organizações para gerenciar a criação de produtos com ênfase na entrega contínua, sem sobrecarregar a equipe de desenvolvimento. Como o Scrum, o Kanban é um processo projetado para ajudar as equipes a trabalharem juntas de maneira mais eficaz.

O Kanban é baseado em 3 princípios básicos:

1. Visualize o que você faz hoje (fluxo de trabalho): ver todos os itens no contexto um do outro pode ser muito informativo
2. Limitar a quantidade de trabalho em andamento (WIP): isso ajuda a equilibrar a abordagem baseada no fluxo para que as equipes não iniciem e se comprometam com muito trabalho ao mesmo tempo
3. Aprimore o fluxo: quando algo termina, a próxima coisa mais alta da lista de pendências é colocada em ação

O Kanban promove a colaboração contínua e incentiva o aprendizado ativo e contínuo e a melhoria, definindo o melhor fluxo de trabalho da equipe possível. Veja como o VersionOne suporta o desenvolvimento de software Kanban.

PROGRAMAÇÃO EXTREMA (XP)

O XP, originalmente descrito por Kent Beck, emergiu como uma das metodologias ágeis mais populares e controversas. XP é uma abordagem disciplinada para fornecer software de alta qualidade de forma rápida e contínua. Promove alto envolvimento do cliente, ciclos rápidos de feedback, testes contínuos, planejamento contínuo e trabalho em equipe próximo para fornecer software de trabalho em intervalos muito frequentes, geralmente a cada 1-3 semanas. A receita original do XP baseia-se em quatro valores simples - simplicidade, comunicação, feedback e coragem - e em doze práticas de suporte:

1. Jogo de Planejamento;
2. Pequenas Versões;

3. Testes de aceitação do cliente;
4. Design simples;
5. Programação em pares;
6. Desenvolvimento Orientado a Testes;
7. Refatoração;
8. Integração contínua;
9. Propriedade do código coletivo;
10. Padrões de codificação;
11. Metáfora;
12. Ritmo Sustentável.

Veja um exemplo de como o XP funciona: Bill cria uma lista de requisitos do cliente fazendo com que este informe “histórias do usuário” que descrevam os recursos. A partir disso, ele constrói um plano de lançamento de software, que será entregue em iterações, com uma entrega a cada duas semanas. A equipe trabalha em pares de programadores, usando reuniões diárias para suavizar os roadblocks. O cliente fornece feedback na forma de mais histórias de usuários. O ciclo se repete até que o software seja entregue.



Benefícios da Metodologia Ágil

Os benefícios do Agile estão ligados diretamente à sua mentalidade mais rápida, mais leve e mais engajada. O processo, em poucas palavras, entrega o que e quando o cliente quiser. Há muito menos tempo desperdiçado na direção errada e todo o sistema é mais rápido para responder às mudanças.

- **Mais rápido:** A velocidade é um dos maiores benefícios da Metodologia Ágil. Um ciclo de vida de desenvolvimento de software mais rápido significa menos tempo entre pagar e receber. Isso, por sua vez, significa um negócio mais lucrativo;
- **Maior satisfação do cliente:** Com o Agile, os clientes não esperam meses ou anos apenas para obter exatamente o que não querem. Em vez disso, eles obtêm iterações de algo muito próximo do que querem, muito rapidamente. O sistema se ajusta rapidamente para refinar a solução de sucesso do cliente, adaptando-se às mudanças no ambiente geral;
- **Valores empregados:** Os funcionários, cujas ideias são valorizadas, são muito mais produtivos do que aqueles que são obrigados a seguir um conjunto de regras. A Metodologia Ágil respeita os funcionários, dando-lhes o objetivo e confiando neles para alcançá-los. Como são eles que estão com as mãos nos controles e aqueles que vêem os obstáculos que surgem todos os dias, os funcionários estão na melhor posição para responder aos desafios e atingir as metas em mãos;

- **Elimina o retrabalho:** Envolvendo o cliente em mais do que apenas as fases de requisitos e entrega, o projeto permanece na tarefa e em sintonia com as necessidades do cliente em cada etapa. Isso significa menos retrocesso e menos tempo fora de contrato entre o momento em que fazemos o trabalho e o tempo que o cliente sugere as revisões.



Continue...

A FM2S espera que o material tenha sido proveitoso para você e que tenha ajudado em entender um pouco sobre as Metodologias Ágeis. Caso tenha interesse, disponibilizamos muitos outros materiais.

Como conhecimento nunca é demais, dê continuidade ao seu aprendizado! Seja um Assinante FM2S e realize o treinamento completo de Metodologias Ágeis e muitos outros!

A Assinatura FM2S possui diversos cursos, entre eles os da formação Seis Sigma. Não perca tempo e se torne já parte desse time!

