

A minimalist line-art illustration in the background. On the right, a person with short hair and round glasses is shown from the chest up, holding a large, thick book with both hands. The book is tilted slightly. In the upper left, a large, faint arc spans across the top, with several small diamond shapes scattered around it. The entire illustration is rendered in a light gray color.

Práticas gerenciais com Scrum

Utilização dos métodos ágeis com base no Scrum para agregação de valor nas organizações.

Prof. Eduardo Montes

Propósito

Habilitar as pessoas e suas organizações a desenvolver seus projetos com base tanto no Scrum, por meio da criação de um ambiente de colaboração, quanto na formação de equipes de alto desempenho sustentadas pelos seus pilares: transparência, inspeção e adaptação.

Objetivos

- Reconhecer os pilares, os valores e os papéis no Scrum.
- Empregar os eventos do Scrum e seus artefatos.
- Aplicar técnicas para o cálculo, o monitoramento e a facilitação dos eventos do Scrum.
- Identificar as melhores e as piores práticas na implementação do Scrum.

Introdução

Em um ambiente altamente competitivo no qual o trabalho remoto ganha a cada dia mais espaço, formar equipes motivadas, auto-organizadas e com um alto desempenho passou a ser uma questão de sobrevivência para as organizações.

O Scrum, o método ágil mais adotado mundialmente, é um dos melhores caminhos para trilhar essa jornada. Neste conteúdo, você aprenderá os componentes necessários para implementá-lo com sucesso.

Iniciaremos nosso estudo pelas razões que levaram à sua criação, detalhando, para isso, seus pilares, os valores a serem incorporados e os papéis exercidos pela equipe, além de seus eventos e artefatos. Complementaremos isso com técnicas comprovadas para planejar, revisar e acompanhar o projeto. Por fim, demonstraremos as principais dificuldades enfrentadas na sua implementação e as formas de solucioná-las.

O Manifesto Ágil

Você sabe o que levou à criação dos métodos ágeis? Quais são as principais diferenças entre os métodos ágeis e os métodos que já existiam anteriormente? A seguir veremos um breve histórico que nos revelará essas respostas.

Apresentação: princípios e características

Na década de 1990, existia uma grande insatisfação na forma com a qual os projetos de desenvolvimento de software eram gerenciados. Tal abordagem, conhecida como cascata por estruturar o projeto em fases sequenciais (em que cada fase depende das entregas da anterior), é ideal para projetos que possuem uma maior previsibilidade, caso dos setores de manufatura e construção, para o qual, aliás, o método ágil foi originalmente criado. As principais reclamações estavam associadas a:

Muitos processos e documentações exigidos.

Complexidade para adaptar às necessidades de mudanças.

Alguns de seus críticos consideravam esse método muito “pesado” para desenvolver softwares devido à sobrecarga de trabalho necessária para gerar toda a documentação. Em oposição a essa abordagem, começavam a surgir na época métodos conhecidos como “leves”, como, entre outros exemplos, o Scrum, o XP (Extreme Programming) e o Crystal.

Em 2001, 17 dos maiores representantes desses métodos mais leves se encontraram para discutir formas de aprimorá-los. A partir desse encontro, foi publicado o Manifesto Ágil. Seus 4 valores e 12 princípios serviram como base para tais métodos, que passaram a adotar o nome “métodos ágeis”.

O Manifesto Ágil expõe seus valores da seguinte forma:



Estamos descobrindo maneiras melhores de desenvolver software, fazendo-o nós mesmos e ajudando outros a fazerem o mesmo. Por meio desse trabalho, passamos a valorizar: 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. Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda

(BECK et al., 2001a)

Como expressa a citação, os valores criados fazem parte de um grande aprendizado prático, analisando as causas dos fracassos e dos sucessos e passando a valorizar aquilo que realmente importa. Descreveremos a seguir outras características desse manifesto:

Indivíduos e interações mais que processos e ferramentas

Por melhores que sejam o processo e as ferramentas usadas, eles não ajudarão muito se as pessoas não estiverem engajadas e não se comunicarem entre si de forma adequada.

Software em funcionamento mais que documentação abrangente

Por melhor e mais detalhada que seja a documentação, de nada adiantará se ela não for transformada em um software funcionando.

Colaboração com o cliente mais que negociação de contratos

É muito mais importante criar um ambiente que favorece a colaboração com o cliente do que investir na negociação com os fornecedores. Os fornecedores são contratados para atender às necessidades dos clientes.

Responder a mudanças mais que seguir um plano

Não adianta nada ter um escopo atendido conforme o planejado se ele não atende mais às necessidades do cliente. Agilidade para adaptar as mudanças é um fator-chave no sucesso do desenvolvimento de software.

Elencaremos agora os princípios do Manifesto Ágil:

Princípios do Manifesto Ágil
Nossa maior prioridade é satisfazer o cliente por meio da entrega contínua e adiantada de software com valor agregado.
Mudanças nos requisitos são bem-vindas, mesmo que tardiamente, no desenvolvimento. Processos ágeis se beneficiam das mudanças visando à vantagem competitiva para o cliente.
Entregar frequentemente um software funcionando de poucas semanas a poucos meses, com preferência para a menor escala de tempo.
Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto ao longo de todo o projeto.
Construa projetos em torno de indivíduos motivados. Dê a eles o ambiente e o suporte necessário, confiando neles para fazer o trabalho.
O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é por meio da conversa face a face.
Software funcionando é a medida primária de progresso.
Os processos ágeis promovem um desenvolvimento sustentável. Patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.
Contínua atenção à excelência técnica e bom design aumentam a agilidade.
Simplicidade (a arte de maximizar a quantidade de trabalho não realizado) é essencial.

Princípios do Manifesto Ágil
As melhores arquiteturas, requisitos e designs emergem de equipes auto-organizáveis.
Em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e, então, refina e ajusta seu comportamento de acordo.

Quadro de princípios do Manifesto Ágil.

Extraído de: Beck et al., 2001b.

O que são os métodos ágeis

Métodos ou frameworks usados para elaborar projetos que compartilham valores e princípios do Manifesto Ágil são considerados métodos ágeis. Kanban, XP, ScrumBan e Scrum são alguns desses métodos disponíveis.

Entre eles, o mais conhecido e usado mundialmente é o Scrum, que detalharemos a seguir. Uma das pesquisas mais famosas do mercado sobre os métodos ágeis, a State of Agile, que já está em sua 15ª edição, mostra o percentual de adoção por método ágil, conforme apresentamos a seguir:

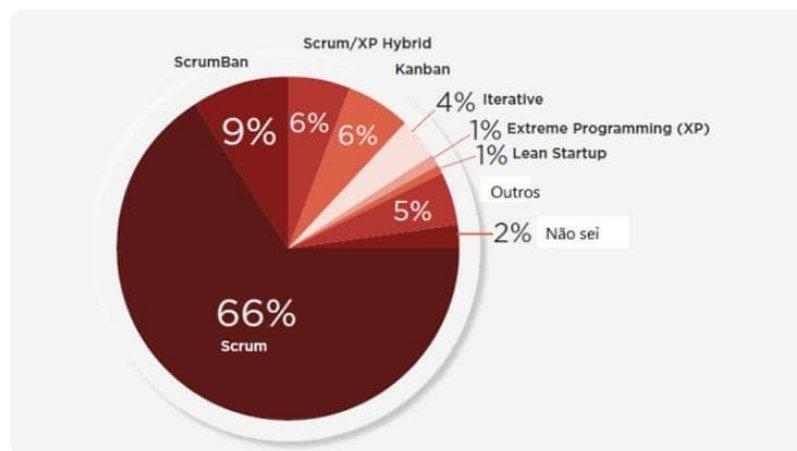


Gráfico: Percentual de adoção por método ágil.

Quando usar os métodos ágeis

Os métodos ágeis foram criados inicialmente para projetos de desenvolvimento de software, porém eles têm sido usados com sucesso em vários outros tipos de projeto.

Eis algumas das características para as quais o uso de métodos ágeis é mais adequado:

Previsibilidade do escopo do projeto

Quanto maior a incerteza daquilo que será feito e maior a necessidade de se adaptar o produto à medida que ele for desenvolvido, mais adequado será o método ágil.

Formas de entrega dos produtos sendo desenvolvidos

Produtos que podem ser entregues de forma incremental e iterativa são adequados ao método ágil.

Riscos

Quanto menor o risco relacionado ao produto, ou seja, quanto menor o impacto relacionado a um problema nas entregas, melhor será o uso do método ágil. Em projetos que envolvem um alto impacto relacionado às suas entregas, como um prédio, uma ponte ou uma indústria, em que a possibilidade de erro pode causar uma catástrofe, deve ser escolhida uma abordagem mais preditiva.

O Scrum

Histórico e definição

Criado por Ken Schwaber e Jeff Sutherland, o Scrum foi apresentado em 1995 na conferência OOPSLA (Object-Oriented Programming, Systems, Languages and Applications). Alinhado com os valores e os princípios do Manifesto Ágil, ele foi desenvolvido inicialmente para projetos de desenvolvimento de software, embora atualmente seja usado em outras áreas, como marketing, pesquisa e inovação.



Time trabalhando com a metodologia ágil.

Segundo Schwaber e Sutherland (2020), o Scrum é um framework leve que ajuda pessoas, times e organizações a gerar valor por intermédio de soluções adaptativas para problemas complexos. Ele é baseado no empirismo, segundo o qual o conhecimento vem da experiência e da tomada de decisão com base no que foi observado, e no *lean thinking*, que significa reduzir o desperdício e concentrar-se no essencial.



Recomendação

Revisite o tópico Quando usar os métodos ágeis para saber se o Scrum é a melhor opção para seu projeto ou sua organização.

Os pilares do Scrum

O Scrum tem como base três pilares:

Transparência

Este pilar está alinhado ao seguinte valor do Manifesto Ágil: indivíduos e interações mais que processos e ferramentas. Para privilegiar as pessoas e suas interações, é essencial que todos estejam cientes das questões mais importantes do projeto. Os artefatos devem ser transparentes, permitindo seu entendimento e sua inspeção.

Inspeção

A inspeção será fundamental para garantir a ágil correção sempre que houver impedimentos ou desvios em relação ao que foi combinado. Implementada por meio de certos eventos, ela leva à necessidade de adaptação, que é nosso próximo pilar.

Adaptação

Quando existem mudanças, desvios, impedimentos ou resultados indesejados, deve-se buscar entender as causas e adaptar o que for necessário para evitar que os problemas sejam repetidos. As pessoas envolvidas precisam ter poder suficiente para agilmente se adaptar por meio dos aprendizados da inspeção.

Os valores do Scrum

Essenciais para o sucesso dos projetos, os seguintes valores do Scrum devem ser incorporados pelo Scrum team:

Compromisso

Todo Scrum team assume o compromisso de alcançar os objetivos traçados — como a meta da sprint — e apoiar uns aos outros.

Foco

A meta e os eventos da sprint, como a daily Scrum, esclarecem as atividades a serem cumpridas no dia e na sprint, direcionando os esforços de cada membro da equipe e mantendo o foco naquilo que precisa ser feito.

Abertura

Todos precisam estar abertos às sugestões dos demais. Alguns eventos do Scrum são propícios para identificar oportunidades de melhoria. Por isso, todos têm de estar prontos para ouvir, agregar e viabilizar essas melhorias.

Respeito

O respeito é fundamental para estimular as interações entre a equipe. Exercer a empatia, se colocando no lugar do outro, melhora a convivência e os resultados da equipe como um todo.

Coragem

É necessário ter a coragem de propor mudanças e de inovar, sem medo de errar. Os membros da equipe precisam assumir responsabilidades e acreditar que são capazes disso.

Scrum team

A base do Scrum é uma equipe de pessoas, o Scrum team, formado por:

Um Scrum master

Um product owner

Alguns developers

O Scrum team também possui várias funções. Seus membros devem ter todas as habilidades e competências necessárias para gerar valor a cada sprint (evento principal do Scrum). Ele é estruturado, autogerenciável ou auto-organizado e empoderado pela organização, ou seja, seus membros têm autonomia e poder suficiente para definir o responsável por cada atividade e como colaborar uns com os outros.

Para Schwaber e Sutherland (2020), o Scrum team deve ser pequeno o suficiente para se manter ágil e suficientemente grande para concluir um trabalho significativo dentro de um sprint. Isso inclui normalmente um grupo de 10 pessoas ou menos. Equipes menores favorecem interações, melhorando a comunicação e a produtividade.



Atenção

Não existem cargos, hierarquia e equipes especializadas no Scrum. Todos do Scrum team são responsáveis pela entrega. O sucesso ou o fracasso é do time, portanto, e nunca de um indivíduo.

Developers

Developers executam as atividades que agregam valor ao produto que está sendo construído, sendo os responsáveis para gerar valor a cada sprint. Por esse motivo, eles devem possuir todas as funções, habilidades e competências necessárias para entregar o produto de forma iterativa e incremental.

Segundo Schwaber e Sutherland (2020), eles são responsáveis por:

Criar um plano para a sprint: o sprint backlog.

Adaptar seu plano diariamente à meta da sprint.



Developer.

Introduzir gradualmente qualidade, aderindo a uma definição de pronto.

Responsabilizarem-se mutuamente como profissionais.

Product owner

O product owner tem um papel estratégico no Scrum team, porque ele é responsável por entender as necessidades dos clientes quanto ao produto a ser desenvolvido, transcrevê-las e priorizá-las a fim de gerar o maior valor agregado dele. Ele faz isso gerenciando o product backlog, que nada mais é do que uma lista priorizada do que é necessário para melhorar o produto.

Scrum master

O Scrum master é o líder-servidor responsável pela eficiência do Scrum team.

Ele é, como o próprio nome diz, o mestre do Scrum e deve:

- Orientar todo o Scrum team para obter o melhor resultado do Scrum com todos os seus componentes.
- Proporcionar um ambiente favorável para a colaboração, para a melhoria contínua e para a autonomia da equipe.



Representação do product backlog.

- Identificar, propor e implementar formas de aumentar a produtividade e o desempenho de todos do Scrum team.
- Ajudar e dar as condições adequadas para o Scrum team incorporar os valores do Scrum.
- Remover qualquer impedimento ao progresso do Scrum team.

Os pilares e o Scrum team

Neste vídeo, apresentamos as principais causas que levaram à criação do Scrum, bem como seus pilares de sustentação, os valores que devem ser incorporados e os papéis exercidos pela equipe.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Vem que eu te explico!

Os vídeos a seguir abordam os assuntos mais relevantes do conteúdo que você acabou de estudar.

O Manifesto Ágil



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Os developers



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Verificando o aprendizado

Questão 1

Qual dos valores a seguir faz parte do Manifesto Ágil?

A

Produtividade e desempenho mais que planejamento e execução.

B

Indivíduos e interações mais que processos e ferramentas.

C

Software em funcionamento menos que documentação abrangente.

D

Contratos bem feitos levam à colaboração com o cliente.

E

Um bom plano reduz a necessidade de mudanças.



A alternativa B está correta.

"Indivíduos e interações mais que processos e ferramentas" é o único valor do Manifesto Ágil entre as opções listadas. Apesar de a produtividade e o desempenho serem preocupações dos métodos ágeis, esse não foi um dos valores incluídos no Manifesto Ágil. As demais são variações com erros dos outros valores.

Questão 2

Qual das situações a seguir é a mais adequada para o uso dos métodos ágeis?

A

Tenho muitas dúvidas das funcionalidades do meu produto e gostaria de construí-lo por módulos e de forma incremental.

B

Quero construir um edifício de 12 andares.

C

Estou participando de uma licitação do governo para construir uma ponte.

D

Minha empresa foi contratada para construir um software que já tem todos os módulos e funcionalidades bem definidas em contrato com multas altas por atraso.

E

Estou iniciando em uma ONG que captou recursos do governo para um projeto com escopo muito bem definido, prestação de contas mensal e devolução integral do investimento no caso de não cumprimento do contrato.



A alternativa A está correta.

Os métodos ágeis se encaixam em um projeto desenvolvido de forma iterativa e incremental. Projetos nos quais o que será feito está muito bem definido e os riscos relacionados são muito altos não são adequados para o uso desses métodos. Nesse caso, o ideal seria uma abordagem de desenvolvimento do projeto mais preditiva.

Eventos do Scrum

Os eventos no Scrum são rituais necessários para garantir o sucesso do projeto. Eles reduzem as necessidades de reuniões complementares e contribuem para manter o Scrum team focado nas entregas. Todo evento Scrum obedece a uma sequência dentro do seu evento maior (a sprint) e tem duração máxima e objetivos específicos.

A sprint

A sprint é o coração do Scrum. Nela ocorrem todos os demais eventos de forma sequencial. A sprint tem início com o evento sprint planning; depois disso, a daily Scrum ocorre todos os dias no mesmo horário e local; e ela finaliza com uma sprint review e uma sprint retrospective.

Após a conclusão da primeira sprint, inicia-se a segunda, e assim por diante, até o final do projeto. Veja a seguir a imagem que representa o ciclo dos eventos do Scrum:



Ciclo dos eventos do Scrum.

Sprint planning

O primeiro evento da sprint é o sprint planning, que tem como objetivo definir o trabalho que será feito na sprint. Sua duração máxima é de oito horas para uma sprint mensal, havendo durações menores para sprints mais curtas.

De acordo com Schwaber e Sutherland (2020), o sprint planning aborda as seguintes questões:

- Por que essa sprint é valiosa?
- O que pode ser feito nessa sprint?
- Como o trabalho escolhido será realizado?

Primeiro, o product owner mostra o que pode agregar mais valor ao produto na sprint. Baseado nisso, o Scrum team colabora para definir a meta da sprint e verificar como ela agregará valor ao produto. Depois, com base na meta definida e nos itens do product backlog priorizados pelo product owner, os developers selecionam os itens que serão feitos. No processo de seleção, o product owner poderá fornecer explicações adicionais e um maior detalhamento.

Apesar da complexidade de selecionar itens realmente concluídos na sprint, alguns fatores-chave aumentam bastante a chance de tal compromisso ser cumprido:

Os developers devem saber claramente o que precisa ser feito para concluir os itens selecionados.

Para cada item do product backlog selecionado, os developers planejam o trabalho necessário decompondo cada item em atividades menores.



Scrum team .

Os developers precisam saber estimar baseando-se no desempenho passado e na capacidade futura.

Construídos no sprint planning, a meta da sprint, os itens selecionados do product backlog e o planejamento para entregá-los compõem o sprint backlog.

Daily Scrum

Definido o que será feito no sprint planning, a daily Scrum tem como objetivo revisar o que será feito no dia e inspecionar o que foi realizado *versus* o que foi planejado, medindo, para tal, o progresso em direção à meta da sprint. Essa inspeção diária pode levar a ajustes no sprint backlog, havendo correções no detalhamento do que foi planejado e até nos itens selecionados.

A daily Scrum ocorre diariamente de forma recorrente no mesmo horário e local para reduzir sua complexidade, com duração máxima de 15 minutos e participação de todos os developers.



Daily Scrum.

Reduz de forma significativa reuniões adicionais.

Aumenta a coesão dos developers.

Agiliza a tomada de decisão, a remoção de impedimentos e a adaptação sempre que necessário.

No final da sprint, ocorrem dois eventos com objetivos distintos: **sprint review** e **sprint retrospective**. Veremos mais detalhes a seguir.

Sprint review

O penúltimo evento da sprint tem como objetivo inspecionar seu resultado e adaptar o que for necessário. Sua duração máxima é de quatro horas para uma sprint mensal, ocorrendo durações menores para sprints mais curtas.

Normalmente, os developers apresentam o que foi realizado e aquilo que foi agregado ao produto. Com base na apresentação, os participantes colaboram para definir os próximos passos.



Saiba mais

Nesse momento, é comum haver ajustes no product backlog relacionados ao detalhamento e à priorização. Tais melhorias servirão de base para o próximo sprint planning.

Sprint retrospective

Trata-se do evento que conclui a sprint. Seu objetivo é identificar oportunidades de melhoria de desempenho e qualidade. A duração máxima é de três horas para uma sprint mensal, havendo durações menores para sprint mais curtas.

O Scrum team inspeciona os resultados da sprint em relação aos indivíduos e às suas interações além dos processos e das ferramentas usados. Os desvios, os impedimentos e suas causas são discutidos, além de serem avaliadas as necessidades de ações preventivas para evitar suas repetições. Identificadas as oportunidades mais úteis, é necessário encaminhá-las e possivelmente incluí-las no sprint backlog.

Recapitulando os eventos, veja o quadro com os objetivos e as particularidades de cada evento:

Evento	Duração máxima e características	Objetivo
Sprint	Um mês.	Criar um incremento no produto que pode ser liberado.
Sprint planning	Oito horas para uma sprint de um mês.	Definir o sprint backlog.

Por quê?

“Por quê? Qual o objetivo a ser atendido com a sprint?”

Inicia-se pela meta da sprint respondendo a essa questão.

O quê?

“O que será feito na sprint para atingir sua meta?”

Para atingir a meta da sprint definida, são selecionados os itens do product backlog que serão feitos na sprint, respondendo à questão.

Como?

“Como será feito cada um dos itens selecionados para atingir a meta da sprint?”

Para cada item selecionado, será feito o detalhamento do trabalho necessário de modo a garantir que eles sejam concluídos dentro da sprint, respondendo à questão.



Atenção

O sprint backlog é de responsabilidade dos developers e serve como foco e compromisso de entrega até o final da sprint, etapa em que ele será revisado e validado na sprint review. No decorrer da sprint, os itens e o detalhamento do sprint backlog são adaptados e refinados.

Incremento

Incremento é a entrega em forma de valor para o produto em desenvolvimento, o qual, por sua vez, precisa atingir a **meta do produto**. Em outras palavras, trata-se da materialização do product backlog em produto a ser consumido por seus clientes.

Um incremento só existe quando ele atende à definição de pronto, que é uma descrição de como garantir que o item que está sendo validado está em conformidade com as medidas de qualidade exigidas para o produto.

Cada sprint gera um novo incremento, que é adicionado aos incrementos anteriores. Juntos, eles devem funcionar de forma integrada, compondo o produto a ser usado pelos seus clientes. A decisão de liberar o incremento e o produto para seus clientes tem de ser feita baseada na meta do produto e na estratégia da organização. Cada sprint deve criar um ou mais incrementos, os quais, por sua vez, são apresentados e validados na sprint review.



Atenção

Em situações específicas, o incremento pode ser liberado independentemente da sprint review.

Recapitulando os artefatos, observe o quadro com seus responsáveis e sua meta:

Artefato	Responsável	Meta a ser atingida
Product backlog	Product owner	Meta do produto
Sprint backlog	Developers	Meta da sprint
Incremento	Scrum team	Definição de pronto

Quadro: Resumo dos artefatos do Scrum.
Elaborado por: Eduardo Montes.

Artefato	Responsável	Meta a ser atingida
Product backlog	Product owner	Meta do produto
Sprint backlog	Developers	Meta da sprint
Incremento	Scrum team	Definição de pronto

Quadro: Resumo dos artefatos do Scrum.
Elaborado por: Eduardo Montes.

Os eventos e os artefatos do Scrum

Neste vídeo, falaremos sobre os artefatos e os eventos do Scrum, e como usá-los para obter sucesso nos projetos.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Vem que eu te explico!

Os vídeos a seguir abordam os assuntos mais relevantes do conteúdo que você acabou de estudar.

O product backlog, o sprint backlog e o incremento



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

A sprint



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Verificando o aprendizado

Questão 1

Qual é o evento do Scrum no qual somente a participação dos developers é obrigatória?

A

Sprint

B

Sprint planning

C

Daily Scrum

D

Sprint review

E

Sprint retrospective



A alternativa C está correta.

A daily Scrum é o único evento que somente a participação dos developers é suficiente. A participação do Scrum master e do product owner é opcional.

Questão 2

Qual é o artefato do Scrum que contém o que deve ser feito para o produto?

A

Product gap

B

Sprint backlog

C

Product backlog

D

Incremento

E

Meta do produto



A alternativa C está correta.

O product backlog contém todos os itens a serem construídos para o produto e é a única origem dos requisitos para qualquer mudança a ser feita nele.

Técnicas complementares ao Scrum

O **Guia do Scrum** não inclui ferramentas e técnicas explicando como os itens de backlog devem ser estimados ou priorizados, tampouco demonstra como os eventos têm de ser conduzidos ou monitorados. Neste módulo, você conhecerá algumas das técnicas mais usadas no mercado e consideradas essenciais para o sucesso do Scrum.



Aplicação de gerenciamento com metodologia Scrum.

Técnicas para priorizar

Iniciemos pelas técnicas para priorizar o product backlog, insumo essencial para selecionar os seus itens, os quais, agregados, gerarão maior valor para o produto. A priorização é de responsabilidade do product owner, enquanto a seleção dos itens do product backlog ocorre na sprint planning, primeiro evento da Sprint, para criar o sprint backlog.



Relembrando

O product backlog e o product backlog são artefatos do Scrum. Já o sprint planning constitui um evento do Scrum.

MoSCoW

MoSCoW é uma técnica muito usada em gestão de projetos para priorizar o escopo, os requisitos e as solicitações de mudanças. Nos métodos ágeis, ele serve para priorizar os itens do product backlog devido à facilidade de uso, à simplicidade e à rápida priorização.

MoSCoW é um acrônimo em inglês usado para classificar cada um dos itens do product backlog em quatro categorias possíveis:

Must have (“deve ter”)

Item obrigatório para o produto (prioritário para a próxima sprint).

Should have (“deveria ter”)

Item muito importante para o produto, mas não obrigatório.

Could have (“pode ter”)

Item desejado, porém não tão importante quanto o *Should have*.

Won't have (“não terá”)

Não será necessário conhecê-lo por enquanto.

Você pode customizar a técnica e até reduzir as categorias para a classificação conforme a necessidade da sua organização.

Uma variação comum é o uso de só duas categorias: must have (obrigatório) e nice to have (desejado).

Análise de decisão envolvendo critérios múltiplos

A análise de decisão envolvendo critérios múltiplos significa selecionar dois ou mais critérios e estipular um peso para cada um dos critérios. Ela retira um pouco da subjetividade da classificação por estabelecer critérios bem definidos e discutidos previamente. Essa análise pode ser usada para:

Priorizar qualquer lista de itens ou de ideias.

Solicitar mudanças, requisitos, projetos, pessoas e itens do product backlog no Scrum.

Técnicas para estimar: Scrum poker

Também conhecido como planning poker, o Scrum poker é uma técnica usada para estimar por meio de um jogo de cartas baseado em consenso (que, aliás, vem ganhando a cada dia mais adeptos nos métodos ágeis). Ele parte da premissa de que quem executa deve ser o mesmo que estima. Os developers são mais comprometidos quando estimam o esforço das suas atividades.

Você se compromete com aquilo que você fala, com sua palavra ou com sua estimativa, mas não o faz com algo sobre o qual não foi consultado e nem sabe se é possível cumpri-lo. Você concorda com isso?

Antes de definir estimativas, há duas atividades que precisam ser realizadas:

1º

Definição da meta da sprint.



2º

Seleção dos itens do product backlog para a sprint.

Para cada item selecionado, detalham-se as atividades necessárias e estima-se o esforço para cada atividade (no momento que se inicia o Scrum poker).

Usaremos então como premissa a hipótese de que já possuímos as atividades relacionadas, aguardando somente suas estimativas. Vejamos agora os passos para aplicar essa técnica.

1

Descrição do item

Primeiramente, o product owner descreve o item a ser estimado.

2

Esclarecimento de dúvidas

Em seguida, os developers o avaliam e tiram suas dúvidas. Item esclarecido para os developers, pode-se iniciar a votação.

3 Seleção da carta

Cada developer seleciona a carta com os pontos de acordo com o esforço necessário para o item baseado em critérios bem definidos.

4

Votação

Cada developer apresenta sua carta de forma simultânea para comparação. Se existir consenso, será possível seguir para o próximo item; caso contrário, inicia-se um debate — normalmente, com as argumentações da maior e da menor carta — e vota-se novamente. O processo se repete até haver um consenso.

Observe a seguir a proposta de seleção para cada carta:

Carta	Explicação
0	Tarefa não precisa ser feita por algum motivo (talvez já esteja pronta).
1	Tarefa muito simples: provavelmente, ela levará menos de uma hora.
2	Tarefa simples: provavelmente, ela durará menos que um turno de trabalho, como uma manhã, por exemplo.
3	Tarefa simples, mas trabalhosa: ela não deve ser subestimada, ocupando pelo menos um turno de trabalho.
5	Tarefa de complexidade mediana, provavelmente tomando um dia de trabalho de um developer se não houver impedimentos.
8	Tarefa complexa: ela vai demandar algum estudo ou muito desenvolvimento, provavelmente tomando alguns dias da semana, como, no máximo, 2 ou 3 dias.
13	Tarefa muito complexa: ela vai demandar um estudo moderado e/ou muito desenvolvimento, levando em média a semana inteira de um developer (cinco dias úteis).
20	Tarefa complexa demais e que não vale a pena ser estimada. Sugere-se quebrá-la em tarefas menores que possam ser estimadas com mais exatidão.
?	Interrogação: não entendi a tarefa. Pode dar mais detalhes?
∞	Infinito: não temos como fazer esta tarefa, ela é longa demais e não cabe em qualquer pipeline de desenvolvimento. Sugiro a quebrarmos em tarefas menores ou dizer ao product owner que não há como fazê-la (mais raro).
Café	Estou cansado de tanto estimar. Vamos tomar um café e depois voltamos a fazer isso?

Quadro: Valores das cartas de votação.

Extraído de: Duarte, 2016, adaptado por Eduardo Montes.

Velocidade

Como todos os itens têm seus pontos calculados, a capacidade de entrega de pontos da equipe (denominada velocidade) é verificada dentro de cada sprint.

A velocidade se torna um parâmetro fundamental para determinar a quantidade de itens a ser incluída em cada sprint.



Exemplo

Se a velocidade da equipe é igual a 100 pontos, não se pode escolher um número de itens, os quais, somados, superem os 100 pontos.

Método Kanban

Importância do Kanban como complemento ao Scrum

A 15ª edição da pesquisa **State of Agile** apresenta uma classificação dos métodos ágeis mais adotados:

1º lugar - Scrum

Presente em 66% das organizações da pesquisa.

2º lugar – ScrumBan

Presente em 9% das organizações da pesquisa.

3º lugar - Kanban

Presente em 6% das organizações da pesquisa.



Comentário

O ScrumBan é o uso do Scrum complementado pelo método Kanban, que será explorado neste tópico.

Criado por Taiichi Ono em 1953 na Toyota, o Kanban é um método de gestão visual do fluxo de trabalho para definir, gerenciar e aperfeiçoar serviços. Simples de usar, ele facilita a visualização do fluxo de trabalho e melhora a comunicação entre os envolvidos.

Kanban é um termo de origem japonesa que significa cartão. Por meio de um quadro denominado quadro Kanban, o cartão vai se transformando no fluxo de trabalho.

Um exemplo mais comum dele é um quadro Kanban para controlar pendências em que a pendência é criada na lista “Para fazer”. Quando se inicia a atividade, movimenta-se para “Fazendo”. Por fim, ao finalizar, a atividade é movimentada para “Feito”.

Quadro Kanban		
Para fazer	Fazendo	Feito

Quadro: Exemplo de quadro Kanban para controlar pendência.

Elaborado por: Eduardo Montes.

Esse quadro pode ser:



Físico

Empregam-se cartões post-its (para representar cada item) que vão se movimentando em cada lista conforme o seu progresso no fluxo.



Virtual

Seu uso ocorre por intermédio de inúmeros softwares disponíveis no mercado.

O quadro Kanban é ideal para dar a transparência ao Scrum team, um dos pilares do Scrum. A visualização do fluxo de trabalho apresenta, de forma clara e fácil, como anda o progresso dos itens de trabalho dos artefatos do Scrum nas seguintes etapas:

- Criação no product backlog
- Desenvolvimento no sprint backlog
- Finalização como um incremento

Analise agora um exemplo de quadro Kanban usado no desenvolvimento de software com o Scrum:

Quadro Kanban							
<i>Product backlog</i>	<i>FaSprint backlog [5]</i>						Incremento
	Desenvolvimento [5]		Teste [5]		Deploy		
	P/ fazer	Fazendo	P/ fazer	Fazendo	P/ fazer	Fazendo	

Quadro: Exemplo de quadro Kanban para desenvolvimento de software com Scrum.

Elaborado por: Eduardo Montes.

Como implementar o Kanban no Scrum

De acordo com Vacanti e Yeret (2018), cada Scrum team deve criar sua definição de fluxo de trabalho contendo os seguintes elementos:

- Definição de como considerar o trabalho iniciado e finalizado.
- Definição das unidades individuais de valor do cliente que estão fluindo por meio do sistema (provavelmente, itens do product backlog).
- Definição do fluxo de trabalho das unidades individuais com cada estágio do início ao fim do fluxo.
- Políticas sobre como o trabalho flui por meio de cada estágio, como a definição de pronto e políticas de transferência entre estágios.
- Definição de como o trabalho em progresso (WIP - *work in progress*) será limitado.
- Expectativa de nível de serviço (SLE - *service level expectation*) definida que comunica uma previsão de quanto tempo deve levar para concluir os itens de trabalho.

Práticas do Kanban para o Scrum

As seguintes práticas do Kanban são essenciais para o Scrum:

Visualização do fluxo de trabalho

- Definir o fluxo acima com seus elementos.
- É necessário definir como e onde ele será apresentado e gerenciado.
- Pode ser em um quadro físico, com o uso de post-its, ou um virtual, implementado em algum software de gestão.

Limitação do trabalho em progresso

- Trabalho em progresso é tudo que foi iniciado e ainda não foi concluído.
- O Kanban trabalha com limitações do trabalho em progresso, concentrando-se nos itens de trabalho em cada um dos estágios do fluxo.
- Os limites de trabalho em progresso podem ser por estágio ou por agrupamento dos estágios.
- Usando o quadro Kanban de exemplo para o desenvolvimento de software apresentado acima, se o limite do estágio desenvolvimento for igual a 5, nunca haverá mais de 5 itens de desenvolvimento em progresso.
- As limitações do trabalho em progresso contribuem para o Scrum teams manter o foco e o comprometimento, além de aumentar a colaboração.

Gestão ativa de itens de trabalho em andamento

Tal gestão ocorre ao se:

- Remover agilmente os impedimentos dos itens de trabalho.
- Garantir que os itens de trabalho serão concluídos de acordo com o nível de serviço esperado.
- Desobstruir itens de trabalho acumulados em um estágio do fluxo.

Inspeção e adaptação da definição do fluxo de trabalho

- A sprint retrospective é uma grande oportunidade para que o fluxo de trabalho e seus elementos sejam revisados.
- Pequenas mudanças podem fazer uma grande diferença no desempenho do Scrum team. As mudanças devem ser sempre avaliadas e adaptadas conforme seus resultados.

Métricas do Kanban

Todas as métricas dispostas a seguir são calculadas tendo como base as definições do fluxo de trabalho e seus elementos detalhados no tópico **Como implementar o Kanban no Scrum**. Incluímos também os nomes das métricas em inglês em parênteses, uma vez que elas são mais conhecidas dessa forma.

Trabalho em progresso (WIP)

Número de itens de trabalho iniciados, mas não concluídos.

Tempo de ciclo (cycle time)

Tempo decorrido do início do item de trabalho até sua conclusão.

Idade do item de trabalho (work item age)

Tempo decorrido do início do item de trabalho até a hora atual.

Taxa de entrega (throughput)

Número de itens de trabalho concluídos por unidade de tempo.

Para calcular tempo de ciclo, é necessário que o Scrum team registre a data e a hora em que se iniciou o item de trabalho e quando ele terminou.

Técnicas para facilitar eventos

Sprint planning

O evento que inicia a sprint tem como objetivo definir o sprint backlog. Para que o sprint backlog seja produtivo, será necessário que, antes do evento, o product owner atualize o product backlog, incluindo a

priorização e o detalhamento dos itens potenciais na seleção para a próxima sprint. Já no sprint planning, é importante que:

- Ele seja facilitado pelo Scrum master.
- Todo o Scrum team participe.
- O product owner inicie explicando os itens de backlog que ele priorizou e tirando as dúvidas dos developers.
- Com o esclarecimento do product owner, os developers iniciem o detalhamento dos itens do product backlog em atividades com suas estimativas.
- No momento que os developers concluírem que já têm trabalho suficiente para a sprint, encerrem-se o sprint backlog e o evento.



Relembrando

As métricas do Kanban funcionam como parâmetros para selecionar quantos itens do product backlog farão parte da sprint. Por exemplo, se o time tem uma taxa de entrega de 10 itens de product backlog por cada sprint, você já sabe que deverá escolher até 10 itens. Nesse caso, também é recomendado incluir um limite de trabalho em progresso de 10 itens para a sprint. A técnica do Scrum poker pode ser usada pelos developers para o detalhamento dos itens do product backlog.

Daily Scrum

Elencaremos agora algumas das melhores práticas em relação à daily Scrum:

- A reunião deve ocorrer sempre no mesmo horário e local, que deve ser de frente para o quadro Kanban.
- Todos ficam de pé durante toda a duração do evento.
- Todos têm de falar e responder às seguintes questões: Todo trabalho planejado na última daily Scrum foi realizado? Existe algum impedimento para a realização do trabalho? Qual trabalho será feito nas próximas 24 horas?

Sprint review

A sprint review ocorre no último dia da sprint para inspecionar o resultado dela, determinando, assim, as adaptações futuras. Para que ela seja produtiva, será necessário que, antes do evento, sejam feitas algumas avaliações por parte dos envolvidos:

person Product owner

Deve avaliar quais clientes podem ajudar na review e deverá convidá-los.

groups Developers

Devem avaliar a melhor forma de apresentar o resultado da sprint.

Já na sprint review, é importante que:

- Ela seja facilitada pelo Scrum master.
- O incremento do produto seja apresentado para apreciação e feedback do product owner e dos clientes presentes.
- O trabalho planejado versus o que foi realizado seja avaliado (itens do sprint backlog concluídos e os não concluídos).
- As métricas sejam revisadas.

Sprint retrospective

A sprint retrospective ocorre logo após a sprint review e tem como principal objetivo analisar os problemas enfrentados e propor melhorias para solucioná-los de forma definitiva. Na sprint retrospective, é importante que:

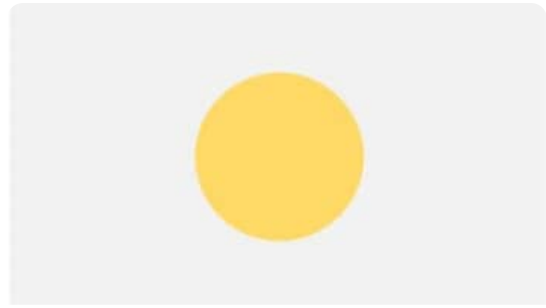
- Ela seja facilitada pelo Scrum master.
- Todo o Scrum team participe e que todos tenham chance de falar.
- Sejam discutidos os principais impedimentos que os developers tiveram e formas de evitar suas repetições nas próximas sprints.
- Os processos sejam inspecionados e adaptados, analisando as métricas do fluxo de trabalho.
- O fluxo de trabalho e seus elementos sejam inspecionados e adaptados para otimizá-los.
- Seja criado um plano de ação ou um registro das mudanças úteis para implementar futuramente.
- Seja selecionada pelo menos uma das mudanças úteis identificadas para implementar na próxima sprint.

Pode-se usar um quadro com três semáforos (verde, amarelo e vermelho) e solicitar que cada membro da equipe escreva em um post-it o que considera relevante para as próximas sprints. Dessa forma:



Verde

Significa que foi bem e deve ser repetido.



Amarelo

Significa que não foi tão bem, mas não é algo crítico.



Vermelho

Significa que certos itens não foram bem e precisam ser melhorados o mais breve possível.

O Scrum master deve conduzir essa sprint para discutir mais os itens marcados como vermelho, gerando ações de melhoria de responsabilidade do próprio Scrum team com a seleção de alguns dos itens para execução na próxima sprint.

Técnicas complementares ao Scrum

Neste vídeo, falaremos sobre a importância de usar técnicas para complementar o Scrum.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Vem que eu te explico!

Os vídeos a seguir abordam os assuntos mais relevantes do conteúdo que você acabou de estudar.

Análise de decisão envolvendo critérios múltiplos



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Quadro Kanban



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Verificando o aprendizado

Questão 1

MoSCoW é uma técnica usada para priorizar os itens do product backlog. Qual das opções a seguir constitui uma de suas categorias?

A

Must have

B

Nice to have

C

Shouldn't have

D

Couldn't have

E

Will have



A alternativa A está correta.

MoSCoW é um acrônimo das suas categorias (*must have*, *should have*, *could have* e *won't have*). Portanto, a única categoria do MoSCoW listada é a *must have*.

Questão 2

Qual das afirmações abaixo sobre Kanban está correta?

A

Kanban foi criado por Peter Drucker em 1975 na Toyota.

B

Kanban" é um termo de origem japonesa que significa lista.

C

Kanban é um método de gestão visual do fluxo de trabalho para definir, gerenciar e aperfeiçoar serviços.

D

Kanban é simples de usar. Ele facilita a visualização do fluxo de trabalho, melhora a comunicação entre os envolvidos e, por isso, deve estar sempre no formato físico.

E

Kanban e Scrum são métodos ágeis, mas funcionam de forma distinta e não se complementam.



A alternativa C está correta.

O Scrum é um framework que necessita de técnicas complementares para ser implementado. O Kanban tem várias métricas e práticas que complementam muito bem o Scrum. Foi até criado um método ágil chamado ScrumBan, que usa o Scrum e o Kanban de forma complementar.

Fases para a implementação

Para haver sucesso na implementação do Scrum, é importante atender às seguintes condições:

Patrocínio da alta direção

É necessário convencer pessoas estratégicas na sua organização para implementar qualquer método ou processo novo.

Resultados consistentes no curto prazo

Quanto mais rápido forem obtidos bons resultados, maior será a credibilidade e menor a resistência das pessoas.

Implantação por fases

A implantação deve evoluir de forma gradual e com visão de longo prazo, para criar uma cultura na organização baseada nos valores e nos princípios do Manifesto Ágil.

Por esses motivos, iniciar com um piloto de escopo limitado e expandir de forma gradual formam uma estratégia que permite um baixo investimento inicial e uma expansão baseada no aprendizado, havendo, em seguida, investimentos adicionais a partir dos resultados apresentados.

Abordagem: montar um Scrum team para um projeto-piloto

Montar um Scrum team a fim de trabalhar em um projeto-piloto para avaliar o Scrum normalmente envolve baixo investimento e menor esforço; por isso, ele é mais fácil de ser aprovado. Além disso, o piloto é uma forma rápida de evidenciar os benefícios do Scrum. A desvantagem de se iniciar por um piloto é que ele não explora todas as possibilidades de uma implementação Scrum.



Scrum team.



Relembrando

Como dissemos anteriormente, para montar um Scrum team, você precisa de um Scrum master, um product owner e de alguns developers (de 3 a 9 developers, dependendo do tamanho do piloto).

Veja algumas dicas para iniciar um projeto-piloto:

1 Comece pelo Scrum master

O Scrum master é o líder servidor, mestre da arte do Scrum e responsável pela eficiência do Scrum team. É fundamental que sua organização tenha alguém muito bem preparado para treinar o Scrum team a fim de que ele saiba obter os melhores resultados do Scrum. Uma consultoria especializada poderá acelerar o processo de aprendizado e a disseminação do Scrum entre o Scrum team.

2

Crie um ambiente propício para a colaboração

O ambiente deve favorecer a transparência e a colaboração entre o Scrum team. Recomendamos a criação de um quadro Kanban físico ou virtual, assim como o uso de ferramentas complementares para facilitar a colaboração.

3

Complemente o Scrum com o Kanban

Defina o fluxo de trabalho com seus elementos.

4

Treine o Scrum team

Capacite todo o Scrum team antes de iniciar o piloto baseado no ambiente de colaboração e no fluxo de trabalho já criado.

5

Procure uma boa consultoria

É importante o apoio de uma consultoria especializada para alavancar os resultados e acelerar a curva de aprendizado.

6

Foque na sprint retrospective

A sprint retrospective talvez seja o evento mais estratégico para o piloto, pois é por meio dela que os pontos de melhoria necessários são identificados. Desse modo, é fundamental que, na reunião, ocorram as seguintes ações:

- Discussão dos principais problemas enfrentados e das mudanças necessárias para corrigi-los.
- Identificação e seleção dos pontos de melhorias para a próxima sprint.

O Scrum é empírico e deve ser adaptado para obter os melhores resultados a cada sprint. A sprint retrospective é a reunião para discutir como os processos, as ferramentas e as métricas poderão ser adaptados de acordo com as necessidades que forem surgindo em cada sprint.

7Expansão

Após a formação do Scrum team e da apresentação dos benefícios do piloto, defina a melhor estratégia para a expansão conforme as necessidades da organização. Faça isso sem perder de vista a criação de uma cultura na organização baseada nos valores e nos princípios do Manifesto Ágil.

Principais barreiras

Tendo como base a 15ª edição da pesquisa State of Agile, os **sete maiores obstáculos** para a adoção dos métodos ágeis são:

Barreira	%
Processos e práticas inconsistentes entre as equipes	46%
Cultura organizacional em conflito com valores ágeis	43%
Resistência geral da organização à mudança	42%
Falta de habilidades/experiência com métodos ágeis	42%
Participação de liderança insuficiente	41%
Apoio de gestão e patrocínio inadequados	40%
Treinamento e educação insuficientes	35%

Quadro: Quais são as barreiras mais significativas para adotar e expandir as práticas Ágile em sua organização atual?

Extraído de: Digital.ai, 2021, p. 12, adaptado por Eduardo Montes.

Para superar esses obstáculos, algumas **estratégias** podem ser adotadas a depender de cada situação:

Patrocínio da alta direção

Sem um patrocínio de peso, alguém que possa trazer os recursos e reduzir as resistências com as lideranças envolvidas nos projetos, a implementação do Scrum tem grandes chances de naufragar nos primeiros obstáculos que aparecerem.

Desse modo, a primeira questão é encontrar pessoas na alta direção para patrocinar e apoiar a implementação. Um bom patrocinador atuará de forma efetiva para que haja a participação suficiente das lideranças e o apoio adequado de outros gestores.

Iniciar com um piloto e expandir gradualmente

Siga as fases para a implementação do Scrum explicadas no tópico anterior.

Contratação de especialista em Scrum

Também é muito importante a contratação de uma boa consultoria especializada em Scrum para treinar as pessoas e ajudar na implementação do método. Especialistas com sucesso comprovado em outras organizações ajudam a reduzir a resistência à mudança e a convencer as pessoas graças aos resultados obtidos internamente ou em outras empresas.

Essa estratégia possui um impacto significativo na maioria das barreiras identificadas. Destacaremos dois exemplos a seguir:

- Falta de habilidades/experiência com métodos ágeis: o especialista habilita as pessoas, trazendo sua experiência e os atalhos para remover os impedimentos e identificar as melhores soluções.
- Treinamento e educação insuficientes: o especialista treina as pessoas de forma adequada.

Interações constantes entre os Scrum masters

É fundamental que exista um compartilhamento entre os Scrum teams das melhores práticas adotadas e dos pontos de melhorias identificados na sprint retrospective. As soluções identificadas devem servir preferencialmente para toda a organização.

É importante haver interações constantes entre os Scrum masters dos Scrum teams para favorecer a definição de processos e de práticas de forma consistente. Provavelmente, os problemas são similares; assim, as soluções adotadas também podem ser semelhantes, favorecendo os ganhos de escala, a curva de aprendizagem e outros ganhos relacionados à padronização. Essa estratégia tem um impacto significativo na barreira denominada **“Processos e práticas inconsistentes entre as equipes”**.

Troca de alguns membros resistentes aos valores ágeis

Uma empresa em que as pessoas são pouco valorizadas e possuem pouca autonomia normalmente terá dificuldades para adaptar-se aos valores e aos princípios do Manifesto Ágil. A maioria delas adapta-se bem aos valores e aos princípios ágeis, porém, dentro da minoria que não se adapta, pode ser necessário haver algumas trocas com o objetivo de demonstrar a importância dos valores ágeis e motivar as pessoas comprometidas.

Monitoramento e divulgação dos resultados de forma consistente

Por meio da interação dos Scrum masters ou até mesmo da criação de um escritório de projetos, é importante haver a divulgação das melhorias obtidas pelos Scrum teams por meio da adoção do Scrum.

O especialista contratado pode, por exemplo, ajudar a fazer essa divulgação ou a estabelecer sua periodicidade. A consistência nos resultados é o melhor remédio para solucionar os conflitos e as resistências, além de evitar a penetração dos métodos tradicionais de desenvolvimento.

Como superar os principais obstáculos na implementação do Scrum

Neste vídeo, falaremos sobre as principais barreiras na implementação do Scrum e como superá-las.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Vem que eu te explico!

O vídeo a seguir aborda o assunto mais relevante do conteúdo que você acabou de estudar.

Inicie com um piloto



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Fases para implementação do Scrum



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Verificando o aprendizado

Questão 1

Qual das alternativas a seguir constitui um dos sete maiores obstáculos na adoção dos métodos ágeis identificados na 15ª edição da pesquisa State of Agile?

A

Processos e práticas inconsistentes entre as equipes.

B

Profissionais não certificados como Scrum master.

C

Treinamento e educação com ênfase nos métodos tradicionais.

D

Cultura organizacional baseada na liderança servidora.

E

Resistência geral da organização às novas tecnologias.



A alternativa A está correta.

Processos e práticas inconsistentes entre as equipes são o maior obstáculo para a adoção dos métodos ágeis segundo a pesquisa. Ter profissionais certificados pode ajudar a implementação do Scrum, porém isso não é um dos principais obstáculos identificados na 15ª edição da pesquisa State of Agile. As demais são variações incorretas dos obstáculos identificados na pesquisa.

Questão 2

Qual das alternativas abaixo é uma das estratégias sugeridas para reduzir os obstáculos (identificados na 15ª edição da pesquisa State of Agile) na adoção de métodos ágeis?

A

Patrocínio da equipe operacional.

B

Contratação de especialista certificado PMP.

C

Patrocinar curso e exame para certificação como Scrum master.

D

Interações constantes entre os gerentes de projetos.

E

Monitoramento e divulgação dos resultados de forma consistente.



A alternativa E está correta.

Monitoramento e divulgação dos resultados de forma consistente são algumas das estratégias sugeridas para reduzir os obstáculos. As demais são variações incorretas das estratégias sugeridas ou não foram estratégias sugeridas.

Considerações finais

Como vimos neste conteúdo, os métodos ágeis foram criados para desenvolver softwares valorizando mais as pessoas e suas interações, assim como o software em funcionamento, a colaboração com o cliente e as respostas ágeis às necessidades de mudanças. Frisamos, por isso, que são muito mais do que métodos: eles constituem, na verdade, princípios e valores que estão revolucionando as organizações.

Método ágil mais conhecido e usado mundialmente, o Scrum fornece um conjunto de pilares, papéis, eventos e artefatos que habilita as pessoas e suas organizações a gerar valor por meio dos seus produtos. Apontamos, contudo, que ele é um framework que precisa de técnicas complementares para ser implementado com sucesso. Entre elas, destacamos o MoSCoW; a análise de decisão, que envolve critérios múltiplos para priorizar os itens do product backlog; o Scrum poker, para estimar os itens de trabalho; e principalmente o método Kanban, que vem sendo muito utilizado para complementar o Scrum.

Informamos também que a implementação do Scrum precisa ser feita de forma gradual, sendo usado, para tal, o empirismo adotado pelo próprio método. Uma boa abordagem é montar um Scrum team para um projeto-piloto. Por isso, demonstramos a importância de se preparar esse projeto de forma adequada, começando pelo Scrum master e criando um ambiente de colaboração para a equipe. Em seguida, deve-se usar o método Kanban para complementar o Scrum e treinar todo o time de forma adequada, sempre tendo como visão de longo prazo a criação de uma cultura na organização baseada nos valores e nos princípios do Manifesto Ágil.

Por fim, descrevemos formas de considerar as principais barreiras identificadas e de buscar estratégias para eliminar ou reduzir seus impactos, apontando ainda como obter um bom patrocinador, contratar um especialista em Scrum para ajudar em toda a sua implementação e criar interações entre os Scrum master, permitindo que suas práticas sejam compartilhadas e aperfeiçoadas, além de monitorar e divulgar os resultados obtidos com a adoção do Scrum.

Podcast

Neste podcast, falamos sobre as práticas gerencias com Scrum. Confira!



Conteúdo interativo

Acesse a versão digital para ouvir o áudio.

Explore +

Acesse a página do Scrum para obter mais detalhes sobre:

- Scrum: faça a leitura completa do **Guia do Scrum**.
- Implementação do Kanban com o Scrum: leia o **The Kanban guide for Scrum teams**.

Referências

BECK, K. *et al.* **Manifesto para desenvolvimento ágil de software**. Manifesto for Agile Software Development. 2001a. Consultado na internet em: 31 ago. 2021.

BECK, K. *et al.* **Princípios por trás do Manifesto Ágil**. Manifesto for Agile Software Development. 2001b. Consultado na internet em: 31 ago. 2021.

DIGITAL.AI. **15th State of Agile Report**. Consultado na internet em: 31 ago. 2021.

DUARTE, L. **Scrum e métodos ágeis** um guia prático. LuizTools, 2016.

SCHWABER, K.; SUTHERLAND, J. **O guia do Scrum** – o guia definitivo para o Scrum: as regras do jogo. Nov. 2020. Consultado na internet em: 31 ago. 2021.

VACANTI, D.; YERET, Y. **The Kanban guide for Scrum teams**. Scrum.org, set. 2019. Consultado na internet em: 31 ago. 2021.