



O Scrum Master Eficaz

Apostila complementar para estudos

Versão 1.0

TIEXAMES www.tandexames.com.br



Histórico de atualizações

Esta apostila se encontra em sua versão **1.0**.



Observações Gerais

Esta apostila faz parte do curso O Scrum Master Eficaz, comercializado e distribuído pela TIEXAMES. Não pode ser vendida separadamente e/ou distribuída em outros canais ou meios que não os oficiais da TIEXAMES.

Índice

1.	O início da sua jornada: O Scrum Master Eficaz.....	8
1.1.	Aviso de marcas registradas e direitos autorais.....	8
1.2.	Boas vindas e recados importantes	8
1.3.	Objetivos desta apostila	8
1.4.	Como extrair o máximo desta jornada?.....	8
1.5.	Conheça seu mentor	9
1.6.	A certificação PSM I da Scrum.Org.....	9
1.7.	Suporte a dúvidas.....	10
2.	Compreendendo a Agilidade.....	11
2.1.	O contexto atual e o porquê de a agilidade ser importante.....	11
2.2.	Ambientes complexos	12
2.3.	Processos tradicionais	13
2.4.	Os problemas com os processos tradicionais	14
2.5.	Processos empíricos	14
2.6.	Processos preditivos vs. processos empíricos.....	15
2.7.	Ambiente vs. tipos de processos.....	15
2.8.	O que é Agilidade?	16
2.9.	O Manifesto Ágil.....	16
2.10.	Os 12 princípios do Manifesto Ágil.....	17
2.11.	Agilidade é um guarda-chuva.....	18
3.	Introdução ao Scrum.....	19
3.1.	O que é o Scrum?	19
3.2.	Principais características do Scrum.....	19
3.3.	Benefícios do Scrum	21
3.4.	Onde o Scrum pode ser aplicado?	21
3.5.	A origem do Scrum	22
3.6.	A anatomia do Scrum	22
3.7.	Visão geral do funcionamento do Scrum	23
3.8.	O Guia Scrum (Scrum Guide).....	24
4.	Conceitos essenciais.....	25
4.1.	O básico sobre Timebox	25
4.2.	ScrumBut e ScrumAnd	26

4.3.	O que é Produto e o que é Valor?.....	26
4.4.	Os três compromissos do Scrum.....	27
4.5.	Compromisso 1: A meta do Produto.....	27
4.6.	Compromisso 2: A meta da Sprint	27
4.7.	Compromisso 3: A definição de Pronto	27
5.	Pilares e Valores do Scrum	29
5.1.	Para que servem os pilares do Scrum?	29
5.2.	Os três pilares do Scrum.....	29
5.3.	Como o Scrum dá vida aos três pilares	30
5.4.	Os valores do Scrum.....	30
6.	Pessoas & Time Scrum	33
6.1.	Os problemas com equipes tradicionais	33
6.2.	O Time Scrum	33
6.3.	O que é ser autogerenciado?	34
6.4.	O tamanho do Time Scrum	34
6.5.	O Time Scrum é multifuncional.....	35
6.6.	O modelo de Tuckman e seus impactos.....	35
6.7.	Times remotos e o Scrum.....	35
6.8.	O Dono do Produto	36
6.9.	A autoridade do Dono de Produto	36
6.10.	O perfil de um Dono de Produto.....	36
6.11.	Os Desenvolvedores.....	37
6.12.	O Scrum Master.....	37
6.13.	A quem o Scrum Master serve?	38
6.14.	Arquitetura do Time Scrum.....	38
6.15.	E o Gerente de Projetos?	38
6.16.	E os gerentes funcionais?.....	39
7.	Gerenciamento Ágil do Escopo	40
7.1.	Gerenciando o escopo de forma ágil	40
7.2.	O Backlog do Produto.....	40
7.3.	A meta do produto.....	42
7.4.	O Backlog do Produto é emergente	42
7.5.	Entendendo os PBIs.....	43
7.6.	Ordenando o Backlog de Produto	43
7.7.	O conceito de preparado	45

7.8.	Refinamento do Backlog do Produto	45
7.9.	A origem da Meta do Produto	46
8.	Sprint & DoD & Timebox	47
8.1.	A Definição de Pronto (DoD)	47
8.2.	A Sprint	48
8.3.	Definindo a duração (horizonte) da Sprint.....	49
8.4.	Antipadrões de uso das Sprints.....	49
8.5.	Os benefícios do Timebox	50
9.	Planejamento da Sprint.....	51
9.1.	Visão geral sobre o Planejamento da Sprint	51
9.2.	O fluxo do Planejamento da Sprint	51
9.3.	Tópico 1 - POR QUE vamos trabalhar nesta sprint?	52
9.4.	Definindo a Meta da Sprint	52
9.5.	Tópico 2 – Em O QUE vamos trabalhar nesta sprint?	53
9.6.	Como definir e gerenciar a capacidade de entrega	53
9.7.	Tópico 3 - Definindo o plano de trabalho (COMO) para a Sprint.....	54
9.8.	O Backlog da Sprint	55
10.	Durante a Sprint	56
10.1.	O que é feito durante a Sprint?.....	56
10.2.	Como o trabalho é realizado durante a Sprint?	56
10.3.	Como lidar com as mudanças durante a Sprint?	57
10.4.	Fuja do Water Scrum Fall	58
10.5.	Gerenciando o trabalho durante a Sprint	58
10.6.	Entendendo o gráfico Burn-down.....	59
10.7.	A reunião diária	59
10.8.	Como realizar a reunião diária?	61
10.9.	Liberação do incremento de produto	62
10.10.	Cancelamento da Sprint	62
11.	Finalizando a Sprint	63
11.1.	Uma visão geral sobre a Revisão da Sprint	63
11.2.	Objetivos da Revisão da Sprint.....	64
11.3.	Retrospectiva da Sprint	65
12.	Planejamento com Scrum	67
12.1.	Uma visão geral sobre Planos e Planejamento	67
12.2.	Mudança de mentalidade	67

12.3.	Múltiplos níveis de planejamento (Planning onion)	68
13.	O básico sobre o Scrum em Escala	70
13.1.	O que é Scrum “normal”?	70
13.2.	O que é Scrum escalado?	70
13.3.	O Dono do Produto em escala	71
13.4.	A Definição de Pronto na escala.....	72
14.	O Scrum Master eficaz	73
14.1.	Scrum Master 1.0	73
14.2.	Scrum Master Eficaz.....	73
15.	Encerramento.....	75

1. O início da sua jornada: O Scrum Master Eficaz

1.1. Aviso de marcas registradas e direitos autorais

Todos os direitos reservados. Nenhuma parte deste material poderá ser reproduzida ou transmitida em/por qualquer meio sem a permissão escrita da TIEXAMES Consultoria e Treinamento Ltda.

A TIEXAMES não licencia o uso de seu material para outras empresas. Se você encontrar outra organização utilizando este material ou parte dele em treinamentos, por favor, denuncie através do e-mail [contato@tiexames.com.br](mailto: contato@tiexames.com.br).

Algumas marcas registradas podem aparecer no decorrer deste curso. O uso delas, bem como de logotipos é apenas para fins editoriais, em benefício exclusivo do proprietário da marca registrada, sem intenção de infringir as regras de sua utilização.

Professional Scrum™, Professional Scrum Master, PSM e PSM I são marcas protegidas da Scrum.org. Esse curso não é endossado e nem afiliado a Scrum.org.

1.2. Boas vindas e recados importantes

Estamos muito felizes que você tenha escolhido **O Scrum Master Eficaz** como instrumento para se tornar um Scrum Master eficaz e obter sua certificação **Professional Scrum Master I da Scrum.Org**.

Acreditamos muito no poder das comunidades e na inteligência coletiva. Para dar vida a estes valores, criamos uma comunidade no **Telegram** e convidamos você a participar também. O link para acesso é: <https://t.me/joinchat/SyOLj3Kmh2IDnET>

Para expandir seus conhecimentos e ficar a par de novidades do mundo Ágil, você também pode se conectar com o seu mentor da Jornada do Scrum Master Eficaz, Kelvin Weiss, no Instagram [@kelvin.weiss](https://www.instagram.com/kelvin.weiss).

Seja muito bem-vindo e uma boa jornada para você!

1.3. Objetivos desta apostila

Esta apostila é um material complementar da jornada do **Scrum Master Eficaz** e deve ser utilizada **em conjunto com as aulas gravadas em vídeo**. Sua leitura é importante para reforçar alguns conceitos vistos durante as aulas, **mas não pode ser usada como única fonte** de preparação para o exame ou para se tornar um Scrum Master eficaz.

O real valor está nas aulas. Sendo assim, o mais importante é que você as assista com muita atenção e só depois leia o conteúdo correspondente na apostila.

1.4. Como extrair o máximo desta jornada?

Cada módulo e aula foi pensado estrategicamente para levar você à meta de ser um Scrum Master Eficaz e obter sua certificação. Aconselhamos não pular nenhum módulo e, a seguir, todas as instruções e dicas que são dadas ao longo das aulas.

Ao assistir às aulas, você pode acelerá-las (velocidade 1.5). Mas tome o cuidado de não acelerar demais a ponto de não dar tempo para o seu cérebro processar as informações. Após cada aula, tome um tempo

para refletir sobre o que você aprendeu. Caso perceba que não esteja se lembrando bem do que foi explicado, é sinal de que você está indo rápido demais. Neste caso, volte para a velocidade 1.

Tenha FOCO. Comece e termine. Muitos profissionais compram o curso, começam e nunca terminam porque se distraem com outras coisas e assim deixam de conquistar importantes metas em suas vidas profissionais. Não seja um procrastinador.

1.5. Conheça seu mentor



+20 anos em TI
Empreendedor aos 17 anos
Métodos e frameworks ágeis desde 2008
MBA em Gestão Estratégica de TI - FGV
Especialização em Gestão de Negócios - FDC
Diversos times e projetos gerenciados
Consultant for Business Software Performance

Redes sociais:

-  @kelvin.weiss (<https://bit.ly/instagramkelvin>)
-  Kelvin Weiss (<https://bit.ly/kelvinweiss>)
-  www.youtube/c/kelvinweiss

1.6. A certificação PSM I da Scrum.Org

Resumo descritivo do exame SCRUM.ORG - Professional Scrum Master I (PSM I) - Baseado no Guide 2020.

Público-alvo: indicado para profissionais da área de desenvolvimento de software que irão atuar no papel de Scrum Master.

Pré-requisitos: nenhum.

Nível de dificuldade: médio. O candidato tem que demonstrar pleno conhecimento do Scrum Guide e de como aplicar as regras em determinadas situações.

Formato do exame: composto por 80 questões de múltipla escolha, sendo necessário acertar no mínimo 85%. São 60 minutos de duração máxima. Pode ser feita consulta ao material, pois não há nenhum tipo de fiscalização durante este exame, porém, o tempo é muito curto, o que torna impossível consultar as respostas para todas as questões.

Idiomas disponíveis: somente em inglês. Devido ao exame ser feito diretamente em uma página web, é possível utilizar o navegador Chrome para fazer uso do tradutor integrado. No Chrome, basta pressionar o botão direito do mouse para acionar a tradução da tela do exame.

Instituto de exame: a certificação PSM I é oferecida pela Scrum.org.

Onde realizar: via internet, em qualquer local e horário (não necessita de agendamento). Basta ter um navegador conectado à internet. Não é necessário ter webcam e microfone, pois este exame não é monitorado.

Certificado: o resultado sai na hora. O certificado digital (em PDF) é liberado imediatamente após a aprovação e vale por tempo indeterminado, não expira e não se faz necessário pagar taxas anuais de manutenção.

Taxa para reexame: se reprovar na primeira tentativa e quiser realizar o exame novamente, terá que ser paga a mesma taxa. Não existe desconto para reexame.

1.7. Suporte a dúvidas

Durante este curso, você terá a oportunidade de entrar em contato com o mentor desta jornada para tirar dúvidas sobre o conteúdo apresentado.

Para interação com o mentor, você deverá utilizar o fórum de discussão dentro do ambiente de ensino virtual. O acesso a este fórum é feito a partir do ícone abaixo:



Dentro do fórum de discussão, você deverá postar suas dúvidas na categoria "**O Scrum Master Eficaz**".

Quando o mentor responder ao seu questionamento, você receberá um aviso por e-mail.

Todas as dúvidas serão esclarecidas em até 2 dias úteis (em geral, as respostas são dadas rapidamente).

2. Compreendendo a Agilidade

2.1. O contexto atual e o porquê de a agilidade ser importante

Nos últimos anos, o mundo mudou consideravelmente. O que era importante perdeu este status e o que sequer era imaginado agora é uma realidade no cotidiano das pessoas. Estamos vivendo um momento em que não é o mais forte que sobrevive, mas literalmente o que se adapta melhor às mudanças. E isso vale para indivíduos e empresas.

Estas rápidas transformações podem ser explicadas por meio do conceito VUCA ou Mundo VUCA.

O termo surgiu nos anos 90, sendo utilizado inicialmente pelas unidades militares dos EUA para designar o novo contexto mundial.

Quatro pilares definem esse novo cenário: **Volatilidade, Incerteza, Complexidade e Ambiguidade**, cujas iniciais em inglês (Volatility, Uncertainty, Complexity e Ambiguity) dão nome ao Mundo VUCA.

Os mesmos fatores listados acima também se tornaram cada vez mais presentes no campo empresarial. No Mundo VUCA, uma companhia no outro lado do planeta ou um grupo de estudantes recém-formados em sua cidade podem tornar um modelo de negócios inteiramente obsoleto em questão de meses.

Para algumas organizações, isso representa um enorme perigo, enquanto outras enxergam no Mundo VUCA a sua maior oportunidade. A diferença entre quem irá conduzir as mudanças e quem será prejudicado por elas está na preparação correta.

Como se preparar para o Mundo VUCA?

As mesmas características que fazem o Mundo VUCA ser tão ameaçador também abrem grandes oportunidades. As tecnologias, processos e dinâmicas de mercado estão aí, e podem ser usadas a favor da sua organização.

Para isso, é preciso ter:

Agilidade

Se as mudanças são rápidas e o amanhã é incerto, agilidade para se adaptar e tomar decisões com a mesma velocidade é um recurso indispensável. Gastar semanas ou meses em planejamento sem ação pode trazer grandes riscos pois, quando um mapa finalmente estiver pronto, a situação já será outra.

Multidisciplinariedade

Como existe uma complexidade entre os muitos componentes do mercado e até mesmo da empresa, é preciso integrar visões diversificadas para dar resposta aos desafios. A liderança puramente vertical precisa dar espaço para ambientes de troca, onde todos podem contribuir com informações e sugestões.

Resiliência

Você, seus parceiros e colaboradores precisam aceitar as transformações potencialmente prejudiciais para os seus negócios. Se uma nova tecnologia foi criada ou um associado importante pediu demissão, é fundamental absorver rapidamente o impacto e voltar para o jogo de cabeça erguida.

Coragem

Embora esta pareça uma característica muito abstrata, a coragem é fundamental para agir em condições ambíguas. Na falta de uma resposta clara, tomar a decisão correta pode ser questão de agir com resolução, aceitando as consequências – positivas e negativas – que resultarem dessa escolha.

2.2. Ambientes complexos

Você já montou um plano para resolver um problema com cada passo detalhado, prevendo cada cenário e de algum jeito ele falha espetacularmente? Até mesmo os planos que você acredita serem os mais infalíveis? Sim, não é? Mas por que isso acontece?

Nossos projetos, planos e produtos frequentemente são sustentados por incertezas que facilmente nos levam a situações ruins. Essas incertezas são provenientes do mundo V.U.C.A (Volátil, Incerto, Complexo e Ambíguo) em que vivemos, o qual anda cada vez mais complexo com o passar dos dias.

Ambientes complexos são basicamente aqueles onde existem variáveis cujo comportamento não é previsível, pois elas mudam o tempo todo, fazendo com que a solução para um problema não seja completamente entendida até que ele esteja resolvido.

Vamos usar o Framework Cynefin para explicar melhor este ponto.

Criado por Dave Snowden em 1999, o Cynefin (pronunciado “quinévin”) é um framework que auxilia na tomada de decisões e no entendimento do mundo atual.

Na visão de Dave Snowden, a dinâmica das coisas pode ser compreendida em 5 domínios:



Claro

Neste domínio, as relações de causa e efeito existem e são previsíveis e repetitivas. É um ambiente governado por Restrições Rígidas e regido pelas melhores práticas. Aqui todos sabem o que, como e quando fazer, o entendimento é claro e difundido, e pode-se associar ao conceito de “checklists”.

Complicado

Neste domínio, também encontramos alta previsibilidade. No entanto, ao contrário do “claro”, em que tudo é simplesmente entendido e executado, aqui precisamos contar com análises de especialistas técnicos que dominam as boas práticas para resolução de problemas.

O complicado é regido pelas Restrições de Governança, ou seja, existem regras que constituem as boas práticas e o que pode ou não ser realizado em determinada função.

Complexo

Ao contrário dos sistemas ordenados, no domínio complexo, temos sistemas imprevisíveis, onde só é possível identificar a relação de causa e efeito em retrospecto. Aqui as necessidades emergem conforme nos deslocamos no tempo e nada é estático e imutável. Precisamos sempre nos atentar ao cenário para adaptação.

Dessa forma, neste domínio, lidamos com Restrições Adaptativas nas quais, conforme as práticas emergem, descobrimos novas formas de fazer as coisas através da experimentação. Se o experimento der certo, nós o difundimos e o amplificamos. Caso haja falha, aprendemos com os erros e pivotamos para a validação de outras hipóteses.

Caótico

Entenda o domínio caótico como uma situação inédita accidental e/ou catastrófica, sendo marcado pela ausência de causa e efeito e restrições. Neste ambiente, precisamos agir rapidamente para estabilizar a situação e restabelecer o “*status quo*”.

Caracterizando-se também pela imprevisibilidade, aqui muitas vezes é necessário introduzir restrições para que possamos sair do caos.

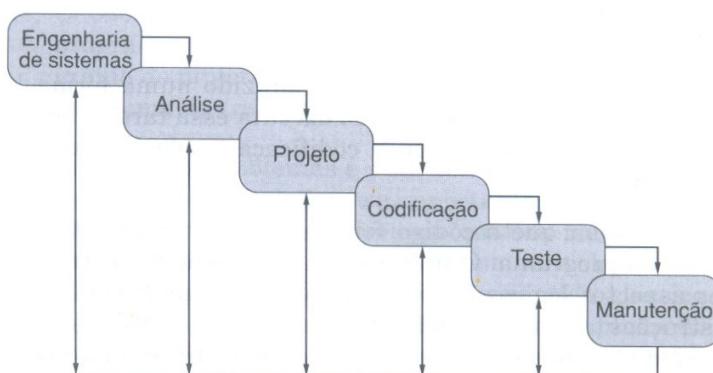
Não-ordenado

Este é o local onde estão aqueles que não sabem em quais dos domínios acima se encontram. É aqui que estamos na maioria das vezes quando não temos uma visão holística da realidade e acabamos interpretando as situações de acordo com nossas preferências ou crenças pessoais.

Se você está com dificuldades em identificar o cenário no qual se encontra, é porque ainda não desmembrou o suficiente para classificar e navegar entre os demais domínios.

2.3. Processos tradicionais

Os processos de desenvolvimento também são conhecidos por processos orientados a planos devido ao rigor na geração da documentação associada. Eles apresentam as seguintes características: abordagem voltada à documentação detalhada da execução das atividades, existência de fases sequenciais, com um conjunto de artefatos para cada fase e ênfase à definição rigorosa de papéis para a execução do trabalho.



A Figura acima representa a metodologia cascata, também conhecida como sequencial ou linear, por se basear em uma sucessão de etapas nas quais uma só é iniciada após o fim da imediatamente anterior a ela. Como se pode ver, o desenvolvimento flui da parte de cima – Engenharia de Sistemas – em direção à manutenção.

Nessa metodologia, inicialmente procura-se compreender completamente o problema a ser resolvido, seus requisitos e suas restrições. Depois planejam-se soluções que atendam a todos os requisitos e restrições para só então iniciar a implementação do projeto. Quando toda a etapa de implementação é concluída, verifica-se junto ao cliente se a solução atende aos requisitos estabelecidos e por fim é efetuada a entrega do produto.

A principal característica deste ciclo de vida é que tudo é definido antes do início do desenvolvimento e há uma clara separação entre os processos. Esta característica de definir tudo antecipadamente e tentar cumprir é chamada de preditivo (*predictive*).

O resultado do planejamento de projeto que usa métodos tradicionais é um **plano detalhado**.

2.4. Os problemas com os processos tradicionais

A abordagem adotada pelo processo cascata acaba trazendo alguns problemas. Entre eles, merece destaque o fato de que os projetos reais dificilmente **seguem o fluxo sequencial**, isto é, o cliente quase sempre **não consegue exprimir todas as suas necessidades** além de ser exigido dele muita paciência, visto que o produto só estará pronto para uso num **ponto tardio do cronograma**. E o maior dos problemas é que se ocorrer um erro em qualquer uma das etapas o resultado pode ser desastroso e frequentemente caro.

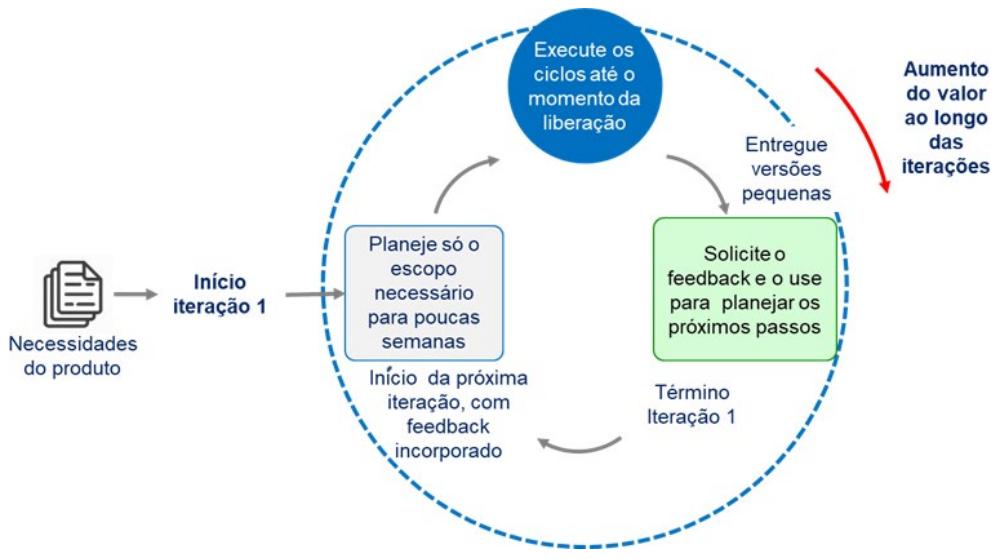
Algumas vezes, o cliente define um conjunto de objetivos gerais que não esclarecem consistentemente os requisitos. Outras vezes, o desenvolvedor não tem certeza da eficiência de parte da solução, da adaptação da aplicação ou mesmo da forma que interação homem-máquina deve ter.

2.5. Processos empíricos

A teoria do empirismo é baseada no conceito que problemas complexos são difíceis de serem previstos, como acabamos de estudar. O Empirismo ajuda as pessoas a tratarem problemas complexos.

Isso requer dar um passo de cada vez, com a realização de uma pequena quantidade de trabalho para adquirir experiência. Veja, como não conseguimos prever tudo o que irá ocorrer, o melhor que podemos fazer é desenvolver um pouco do produto, inspecionar este trabalho – ou seja, analisar se estamos no caminho e com base nesse feedback ajustar os planos – e só então iniciar novo ciclo, desenvolvendo mais um pouco de produto.

A ideia aqui é dividir para conquistar. Quebrar o desenvolvimento do produto em pequenos blocos para que fiquem mais gerenciáveis e fáceis de serem entendidos e absorvidos.



Vejamos aqui na ilustração para entender um pouco melhor isso: tudo parte das necessidades do produto, mas você não precisa ter tudo detalhado, apenas o suficiente para começar o desenvolvimento. Imagine que, se você tiver 100 itens, uns 10 deles estarão detalhados, uns 20 ou 30 estarão com algum nível de detalhe e os demais estarão apenas com o nome.

A partir das necessidades do produto, ou seja, do escopo, iniciamos a Iteração 1. Uma Iteração é um ciclo completo de desenvolvimento. Para essa Iteração, você vai planejar só o escopo necessário para poucas semanas, normalmente menos que 1 mês. Portanto, trata-se de um horizonte curto, mais controlável e gerenciável.

Você então desenvolve estes requisitos/itens durante estas poucas semanas. Ao final deste período, entrega ao cliente uma versão pequena com estes requisitos, solicita feedback e o usa para planejar os próximos passos. Então você inicia a próxima iteração já com o feedback incorporado e neste próximo ciclo certamente terá um desempenho melhor que no primeiro.

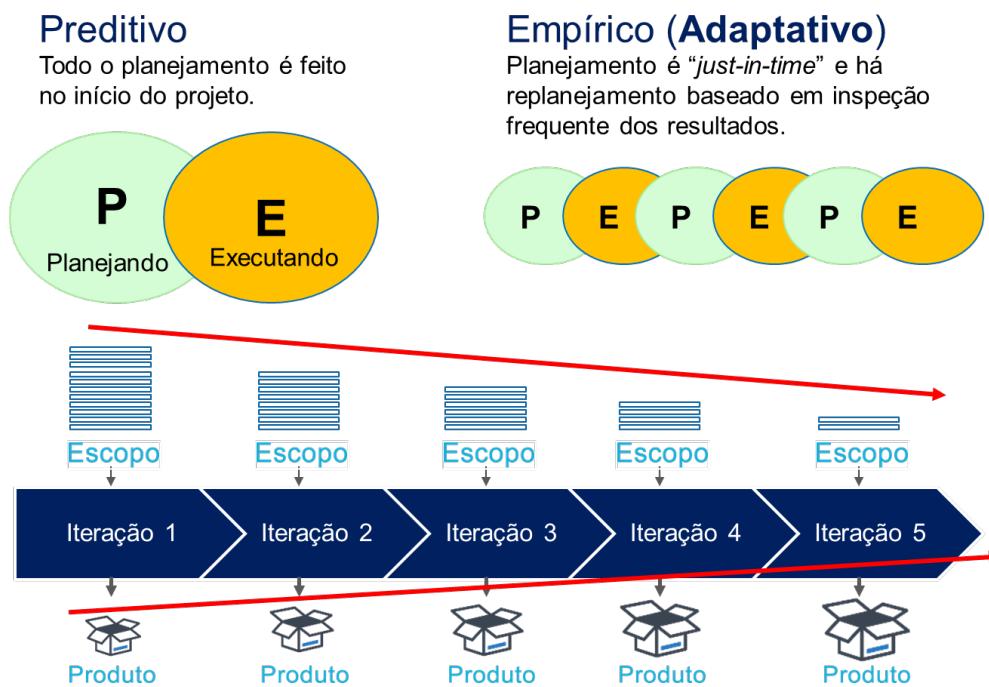
E assim você vai penejando, entregando, pedindo feedback e ajustando os rumos dos trabalhos.

Isso faz com que haja uma redução de riscos ao longo das iterações porque, conforme você vai fazendo, o cliente já vai dando feedback e dizendo se estamos ou não no caminho. A chance de falhar é muito menor.

2.6. Processos preditivos vs. processos empíricos

Veja que no modelo preditivo, como é o caso do modelo cascata, todo o planejamento é feito no início do projeto. Temos uma grande fase de planejamento e depois outra grande fase de execução. Primeiro você planeja tudo e depois executa tudo.

No modelo empírico, o planejamento é “just in time” e há replanejamento baseado em inspeção frequente dos resultados: você planeja um pouco de trabalho, depois o executa, avalia o resultado obtido, ajusta o que tiver que ajustar e inicia um novo ciclo, planejando mais um pouco de trabalho já incorporando o aprendizado que teve no ciclo anterior.



2.7. Ambiente vs. tipos de processos

Quanto temos um ambiente onde requisitos estão bem acordados e definidos e conhecemos bem a tecnologia que será empregada no desenvolvimento, então temos um cenário mais simples. Quando temos um ambiente onde requisitos e tecnologia estão longe da certeza e do acordo, temos um ambiente anárquico.

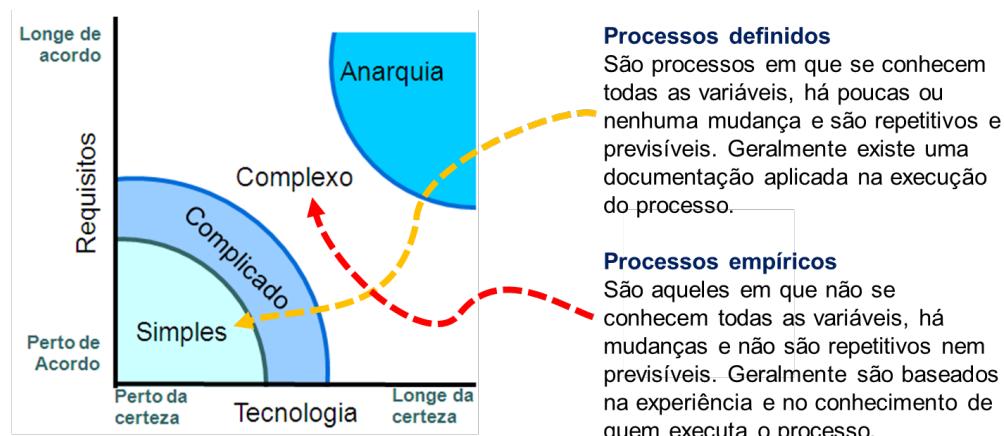
Talvez estejamos a fazer coisas que ainda não existam ou ainda não existe tecnologia disponível. Por exemplo, fazer um robô com Inteligência Artificial: não temos nem os requisitos definidos e nem tecnologia para isso.

Existe uma área onde o estabelecimento dos requisitos e da tecnologia flutua entre a definição e a indefinição. É a região onde temos ambientes complexos. O mesmo acontece com a tecnologia.

Então, de forma geral, existem dois tipos de processos: Processos definidos e Processos empíricos.

Processos definidos são aqueles em que se conhecem todas as variáveis, as quais têm pouca ou nenhuma mudança ao longo do processo, tornando-os repetitivos e previsíveis. Geralmente existe uma documentação aplicada à execução do processo e um bom local para aplicá-lo é numa linha de produção. Ou seja, o processo definido é empregado para atividades que ficam na região “simples”.

Os processos empíricos são os que não se conhecem todas as variáveis, havendo mudanças ao longo do processo, fazendo com que não sejam repetitivos e previsíveis. Geralmente são baseados na experiência e no conhecimento de quem os executa. Ou seja, o processo empírico é aplicado para atividades que ficam na região “complexo”.



O empirismo afirma que o conhecimento vem da experiência e devemos tomar decisões com base no que se conhece.

2.8. O que é Agilidade?

Então, o que é ser ágil?

A palavra "ágil" é um adjetivo que significa *mover-se rápida e facilmente ou pensar rapidamente e de uma forma inteligente*. Ser ágil é ter rapidez, desembaraço, eliminando a burocracia desnecessária. Ser ágil é entregar aquilo que gera o maior valor para o negócio do cliente, de forma cada vez mais rápida e com qualidade.

Ser ágil é fazer as coisas complexas de forma simples. Ser ágil é ter a equipe toda comprometida com os objetivos do produto ou iniciativa. E o mais importante: ser ágil é ter **capacidade de responder rapidamente a mudanças**. Nada mais constante que a mudança, portanto, esteja preparado para ela.

2.9. O Manifesto Ágil

Apesar de algumas práticas e métodos ágeis existirem há mais de 20 anos, foi em 2001 que o pontapé foi dado para a grande disseminação das abordagens ágeis.

Nesta época, os principais responsáveis pela teoria e pela aplicação de diversos métodos se reuniram para discutir sobre as melhores maneiras de gerenciar e desenvolver softwares. Entre os participantes deste grupo podemos destacar nomes como Kent Beck, Ken Schwaber e Jeff Sutherland.

Destes encontros acabaria por sair o Manifesto Ágil e a formação da Aliança Ágil, uma organização sem fins lucrativos destinada a promover a adoção dos métodos ágeis.

O Manifesto Ágil documentou os princípios orientadores para o desenvolvimento ágil e definiu uma filosofia em torno de um conjunto de metodologias existentes. Nele está escrito o seguinte:

Estamos descobrindo maneiras melhores de desenvolver software, fazendo-o nós mesmos e ajudando outros a fazê-lo. Através deste trabalho, passamos a valorizar:

Indivíduos e interação entre eles 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.

Além desta declaração, o Manifesto Ágil traz ainda **12 princípios** que devem ser seguidos por todos que queiram ter agilidade

2.10. Os 12 princípios do Manifesto Ágil

01 - Nossa maior prioridade é satisfazer o cliente através da entrega contínua e adiantada de software com valor agregado.

02 - Mudanças nos requisitos são bem-vindas, mesmo tardivamente no desenvolvimento.

03 - Processos ágeis tiram proveito das mudanças, visando à vantagem competitiva para o cliente.

04 - Entregar frequentemente software funcionando, de poucas semanas a poucos meses, com preferência à menor escala de tempo.

05 - Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto.

06 - Construa projetos em torno de indivíduos motivados. Dê-lhes o ambiente e o suporte necessários e confie neles para fazer o trabalho.

06 - O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é através de conversa face a face.

07 - Software funcionando é a medida primária de progresso.

08 - Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.

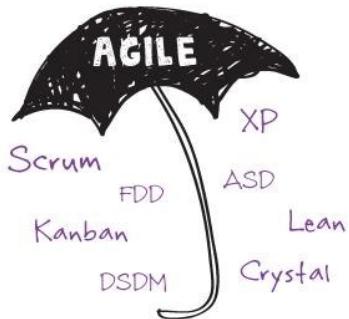
09 - Contínua atenção à excelência técnica e ao bom design aumenta a agilidade.

10 – Simplicidade – a arte de maximizar a quantidade de trabalho não realizado – é essencial.

11 - As melhores arquiteturas, requisitos e designs emergem de equipes auto-organizáveis.

12 - Em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e então refina e ajusta seu comportamento de acordo.

2.11. Agilidade é um guarda-chuva



O movimento ágil comprehende vários frameworks e métodos, é como se fosse um guarda-chuva que define características gerais que os modelos ágeis devem conter.

Atualmente existe uma vasta quantidade de metodologias, frameworks e técnicas ágeis disponíveis, como por exemplo: Scrum, Extreme Programming, FDD, TDD.

É possível também você combinar as práticas destes modelos, pois eles não são excludentes. Por exemplo, você pode combinar as práticas do framework Scrum com a metodologia Extreme Programming, adicionando ainda a prática do TDD.

Você também pode adotar na sua empresa apenas um dos modelos, como o Scrum. Mas, depois de um tempo de uso e conforme você ganha a experiência, verá que, para conseguir todo o benefício do Scrum, pode ser necessário adicionar práticas ágeis de outras metodologias. Por exemplo, você pode trabalhar com o Scrum sem ter testes automatizados no início, mas se usá-los, vai conseguir extrair muito mais valor do Scrum.

Então métodos e práticas ágeis não são excludentes, eles podem ser combinados sempre que necessário, desde que isso gere mais valor para o seu cliente em menos tempo.

3. Introdução ao Scrum

3.1. O que é o Scrum?

O Scrum é uma estrutura (framework) leve que ajuda as pessoas, equipes e organizações a gerar valor através de soluções adaptativas para problemas complexos. Em poucas palavras: Scrum requer um Scrum Master para promover um ambiente onde:

1. Um Dono de Produto encomenda o trabalho para um problema complexo num Backlog do Produto.
2. A Scrum Team transforma uma seleção do trabalho num Incremento de valor durante um Sprint.
3. A Scrum Team e os seus stakeholders inspecionam os resultados e se ajustam para o próximo Sprint.
4. Repete.

O Scrum é um framework dentro do qual pessoas podem tratar e resolver **problemas complexos e adaptativos**. Ele não é um processo detalhado e prescritivo para todas as situações, por isso é chamado de um framework e não de uma solução completa. É fácil de ser entendido e difícil de ser implementado.

O Scrum adota uma abordagem empírica, aceitando que o problema não pode ser totalmente entendido ou definido, focando na maximização da habilidade do time de responder de forma ágil aos desafios emergentes.

3.2. Principais características do Scrum

O Scrum utiliza o modelo Iterativo e Incremental para desenvolvedor produtos complexos.

A utilização desse modelo está nas bases dos métodos ágeis, como o Scrum. A ideia é que a criação de um produto seja pautada por vários ciclos curtos, em que funcionalidades são introduzidas, feedbacks coletados e requisitos revistos.

Assim, é possível atingir um maior nível de satisfação do cliente e garantir que o resultado final esteja dentro do esperado. Se, no modelo em cascata, a evolução é feita como um processo contínuo e sequenciado, no desenvolvimento iterativo e incremental, a empresa diminui tarefas e repete etapas sempre que for necessário.

Pretende-se, com isso, reduzir o número de falhas na solução entregue ao usuário, criando um ambiente de trabalho que seja mais prático e capaz de fazer mudanças em todas as etapas de desenvolvimento.

Incremental



Iterative



Um processo **iterativo** é aquele que faz progresso através de tentativas sucessivas de refinamento. Por exemplo, uma equipe faz sua primeira tentativa para construção de um produto, porém existem pontos de informação falhos ou incompletos em algumas (talvez muitas) partes. A equipe de forma **iterativa** refina essas partes até que o produto atinja o ônus de satisfatório. Com cada **iteração**, ele é melhorado através da adição de mais e mais detalhes.

Um processo **incremental** é aquele em que o produto é construído e entregue por pedaços. Cada pedaço ou **incremento** representa um subconjunto de funcionalidades completas. O **incremento** pode ser pequeno ou grande. Cada **incremento** é totalmente produzido, e a expectativa geral é que o trabalho tenha a conclusão mais completa possível.

O time Scrum é autogerenciado.

O Scrum é construído com base na inteligência coletiva das pessoas que o utilizam. Em vez de fornecer instruções detalhadas às pessoas, as regras do Scrum guiam as suas relações e interações. Há vários conceitos para o que é um time autogerenciado, mas do ponto de vista do Scrum, um time autogerido significa que é o próprio grupo que decide internamente quem faz o que, quando e como.

O cliente é parceiro.

Os princípios ágeis incluem requisitos de manutenção e documentação leve e reconhecem que a mudança é uma realidade normal e aceitável no desenvolvimento de produtos e soluções complexas. Isso faz com que seja necessária uma colaboração estreita entre pessoas de negócio e técnicas.

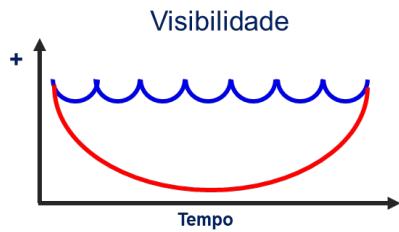
Para o Scrum, o cliente é um parceiro e deve ter envolvimento ativo ao longo de todo o trabalho, ao contrário dos modelos tradicionais nos quais o cliente participa normalmente no início do levantamento dos requisitos e depois só reaparece na entrega.

Planejamento “just in time”.

No Scrum, o planejamento é “just in time” e há replanejamento baseado em inspeção frequente dos resultados. Você planeja um pouco de trabalho, o executa, avalia o resultado obtido, ajusta o que tiver que ajustar e inicia um novo ciclo, planejando mais um pouco de trabalho, já incorporando o aprendizado do ciclo anterior.

3.3. Benefícios do Scrum

O Scrum proporciona maior visibilidade porque adota o conceito de entregas menores e mais frequentes, conforme ilustração.

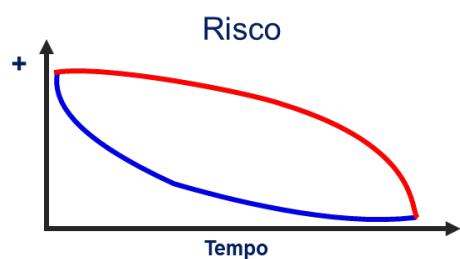


Como o planejamento e detalhamento do escopo é feito gradualmente, existe uma maior habilidade para mudanças e consequente adaptação.

A entrega de valor é feita de forma mais rápida, uma vez que pequenos incrementos já são entregues logo no início dos trabalhos.



E por fim há uma diminuição dos riscos logo no início do projeto, pois desde cedo partes do trabalho já são entregues ao cliente e é possível identificar se o caminho adotado é o mais adequado.



█ Scrum █ Método cascata

3.4. Onde o Scrum pode ser aplicado?

O foco do Scrum está em prover um método que deixa clara a eficácia relativa das práticas de gerenciamento de produto e técnicas de trabalho que você usa, de modo que possa continuamente melhorar o produto, o time e o ambiente de trabalho. Ou seja, o Scrum deixa claro o quanto bem o time faz suas atividades em seu dia a dia e o ajuda a melhorar.

O Scrum não vai dizer como você tem que desenvolver um produto, um hardware ou algo inovador, mas vai lhe dar mecanismos para saber se está fazendo bem o seu trabalho e melhorar continuamente. Ele foi inicialmente desenvolvido para gerenciar e desenvolver produtos.

Iniciando no começo dos anos 90, o Scrum tem sido usado extensivamente, mundialmente, para:

1. Pesquisar e Identificar mercados viáveis, tecnologias e funcionalidades de produtos;
2. Desenvolver produtos e melhorias;
3. Liberar produtos e melhorias frequentes, chegando a várias vezes por dia;
4. Desenvolver e sustentar a Nuvem (online, segura, sob demanda) e outros ambientes operacionais para uso de produtos;
5. Sustentar e renovar produtos.

Scrum tem sido usado para desenvolver **software, hardware, software embarcado, redes de funções interativas, veículos autônomos, escolas, governo, marketing**, gerenciar a operação da organização e quase tudo que usamos no dia a dia nas nossas vidas, como indivíduos e sociedade.

Como tecnologia, mercado, complexidades ambientais e suas interações têm aumentado rapidamente, a utilidade do Scrum em lidar com a complexidade é provada diariamente. Ele demonstra efetividade especialmente na transferência de conhecimento iterativo e incremental.

3.5. A origem do Scrum

Ken Schwaber & Jeff Sutherland criaram e vêm evoluindo o Scrum desde a década de 90. Este nome vem de uma formação do Rugby, um esporte coletivo originário da Inglaterra.

Scrum ou formação ordenada é uma situação frequente no rugby, geralmente usada após uma jogada irregular ou em alguma penalização. Nesta formação, 8 jogadores de cada time devem se encaixar para formar uma muralha, isto é, é muito importante que seja realizado um trabalho de equipe, pois, se um dos jogadores na formação falhar, toda a jogada será comprometida.

3.6. A anatomia do Scrum

O Scrum tem uma anatomia conhecida como 3-5-3: 3 responsabilidades (papéis), 5 eventos e 3 artefatos.

Responsabilidades

- Dono de Produto: é responsável por maximizar o valor do produto resultante do trabalho do Time Scrum.
- Scrum Master: é responsável pela implementação do Scrum tal como definido no Guia do Scrum.
- Desenvolvedores: são as pessoas do Time Scrum que estão empenhadas em criar qualquer aspecto de um Incremento utilizável em cada Sprint.

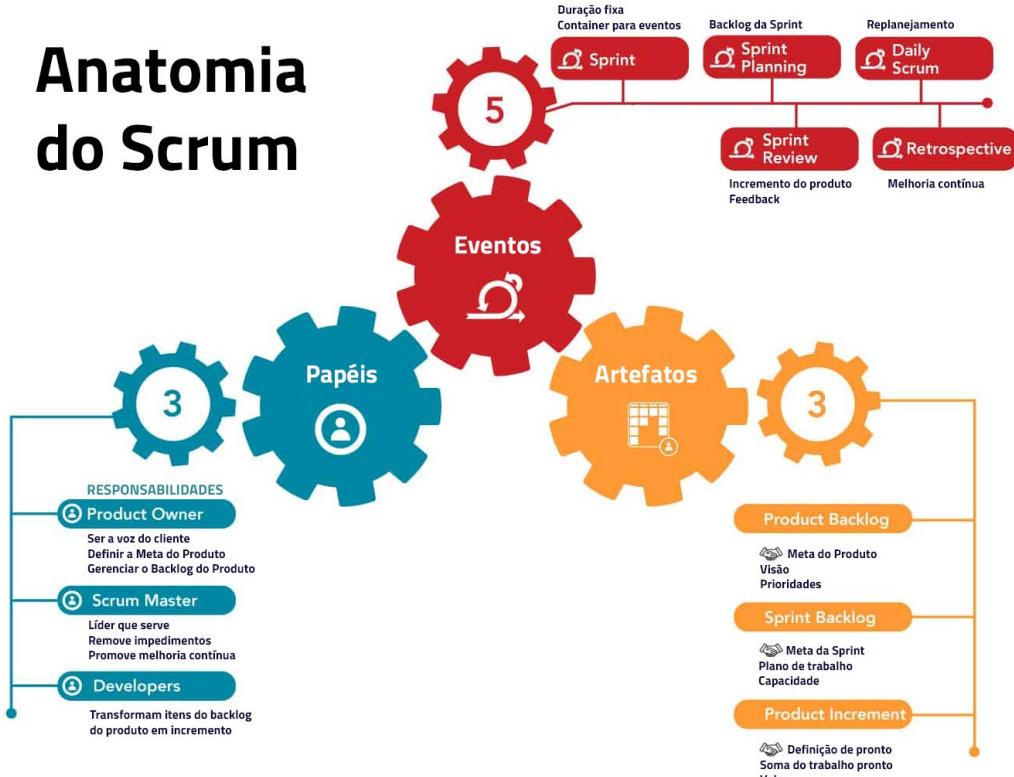
Eventos

- Sprint: são eventos de duração fixa de um mês ou menos para criar consistência. Podemos dizer que são as iterações.
- Planejamento da Sprint: é onde é determinado qual será o trabalho a ser realizado.
- Daily Scrum (Scrum diário): é um evento de 15 minutos para os Desenvolvedores com o objetivo de inspecionar o progresso rumo ao Sprint Goal e adaptar o Sprint Backlog conforme necessário, ajustando o trabalho planeado que se aproxima.
- Revisão da Sprint: é o momento de inspecionar o resultado da Sprint e determinar adaptações futuras, com foco no produto que foi entregue.
- Retrospectiva da Sprint: planejar formas de aumentar a qualidade e a eficácia. É um evento para gerar melhoria contínua.

Artefatos

- Backlog do Produto: é uma lista emergente e ordenada do que é necessário para melhorar o produto. É a única fonte de trabalho levada a cabo pelo Time Scrum.

- Backlog da Sprint: é um plano de e para os Desenvolvedores. O Backlog da Sprint é composto pela Meta da Sprint (porque), o conjunto de itens selecionados do Backlog do Produto para a Sprint (o que), bem como um plano acionável para a entrega do Incremento (como).
- Incremento do Produto: representa aquilo que foi produzido ao longo das sprints. Cada Incremento é acrescentado a todos os Incrementos anteriores e cuidadosamente verificado, assegurando que todos eles funcionem em conjunto. A fim de fornecer valor, o Incremento deve ser utilizável.



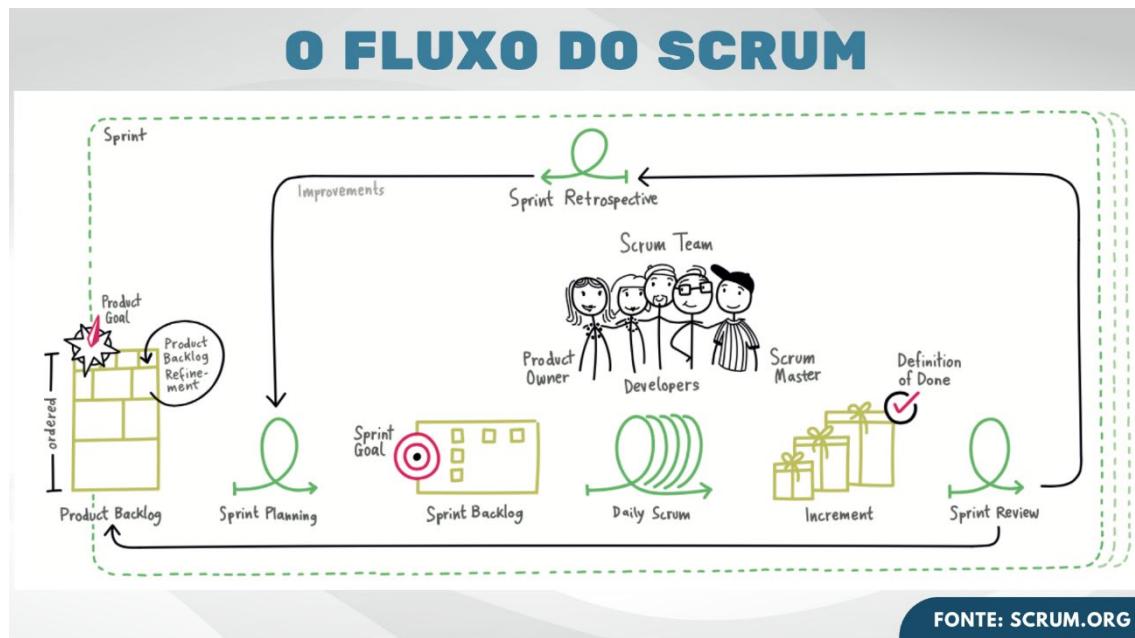
Fonte: Scruminc

3.7. Visão geral do funcionamento do Scrum

Tudo se inicia com a construção de um Backlog de Produto minimamente ordenado e refinado. A partir deste ponto, o Time Scrum se reúne para planejar o escopo da Sprint atual. Isso é feito em um evento chamado Planejamento da Sprint, onde são discutidos por que a sprint será feita, o que será feito e como será feito. A saída desta reunião é o Backlog da Sprint.

Após o planejamento, o trabalho se inicia e diariamente os Desenvolvedores se reúnem para inspecionar e adaptar o trabalho. Este encontro chama-se Reunião Diária.

No último dia da Sprint ocorre a Revisão da Sprint, na qual o incremento de produto é inspecionado e o backlog de produto adaptado. Na sequência, inicia-se a Retrospectiva da Sprint na qual são discutidas ações de melhoria tanto no processo quanto nas relações do time.



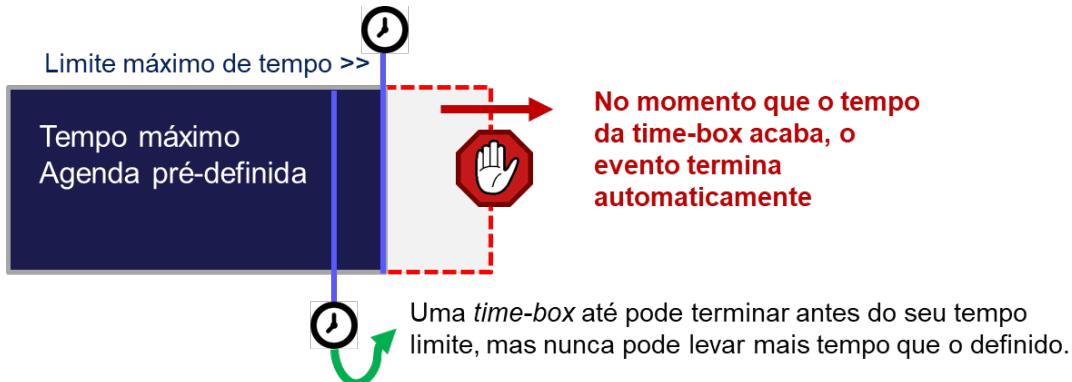
3.8. O Guia Scrum (Scrum Guide)

Desenvolvido e mantido pelos criadores do Scrum, Ken Schwaber e Jeff Sutherland, o Guia do Scrum (Scrum Guide) é o primeiro manual de consulta sobre o Scrum. No documento, as regras e o funcionamento do Scrum estão descritos de forma bem concisa e objetiva.

Para ler o Scrum Guide, acesse: <https://scrumguides.org/>.

4. Conceitos essenciais

4.1. O básico sobre Timebox



Timebox é uma das regras do Scrum. TimeBox é um período máximo. É um tempo fechado, imutável, que tem uma duração máxima. Veja na ilustração, onde temos o retângulo em azul representando uma quantidade de tempo X, que é o limite, retratando um timebox ou uma caixa de tempo. Veja também que uma das regras do timebox é que ele até pode terminar antes do seu tempo limite, mas nunca levar mais tempo que o definido. Nós temos que nos adaptar para executar as atividades no tempo disponível de cada timebox.

Agora observe também que você não pode estender esse tempo máximo. No momento em que o tempo do timebox acaba, o evento termina automaticamente.

Vamos dar alguns exemplos para facilitar a compreensão: a reunião diária do Scrum é um timebox de 15 minutos e por isso não pode levar mais que esse tempo. E se chegar ao final dos 15 minutos e não tiver dado tempo de concluir a reunião? Ela termina do mesmo jeito e o time deve tomar ações corretivas para que na próxima oportunidade dê tempo.

E se os assuntos já tiverem sido tratados, não restando mais nenhum assunto pra tratar, com apenas 10 minutos de duração? Nesse caso, a reunião diária pode ser finalizada sem problema nenhum.

No Scrum, os eventos de Planejamento da Sprint, Reunião Diária, Revisão da Sprint e Retrospectiva da Sprint são eventos timeboxed. A seguir, veja uma tabela com o timebox de cada evento.

EVENTOS SCRUM X TIMEBOX				
SE A SPRINT FOR DE:				
EVENTO	UM MÊS	3 SEMANAS	2 SEMANAS	1 SEMANA
PLANEJAMENTO DA SPRINT	8 HORAS	Geralmente menos que 8 horas (~6 horas)	Geralmente menos que 8 horas (~4 horas)	Geralmente menos que 8 horas (~2 horas)
REUNIÃO DIÁRIA		15 MINUTOS		
REVISÃO DA SPRINT	4 HORAS	Geralmente menos que 8 horas (~3 horas)	Geralmente menos que 8 horas (~2 horas)	Geralmente menos que 8 horas (~1 horas)
RETROSPECTIVA DA SPRINT	3 HORAS	Geralmente menos que 8 horas (~2h15)	Geralmente menos que 8 horas (~1h30)	Geralmente menos que 8 horas (~45 min)

O tempo entre parênteses é o tempo esperado, dado que a sprint é menor que 30 dias.

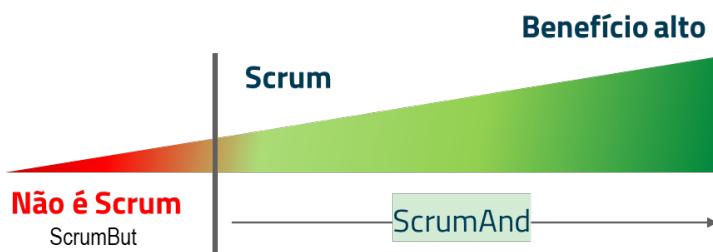
FONTE: SCRUM GUIDE

4.2. ScrumBut e ScrumAnd

SCRUMBUT são as razões pelas quais as equipes não podem aproveitar ao máximo o Scrum para resolver seus problemas e aproveitar todos os benefícios do desenvolvimento de produtos usando o Scrum. Cada papel, regra e timebox do Scrum é projetado para fornecer os benefícios desejados e resolver problemas recorrentes previsíveis.

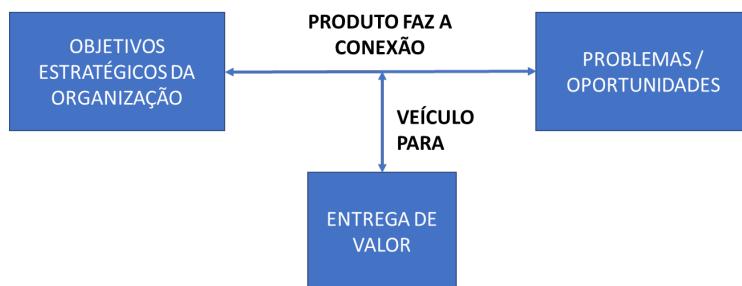
"ScrumBut significa que o Scrum expôs uma disfunção que está contribuindo para o problema, mas ela é muito difícil de corrigir." Um ScrumBut mantém o problema enquanto modifica o Scrum para torná-lo invisível (o problema é varrido para debaixo do tapete).

O ScrumAnd é quando práticas são adicionadas sobre o Scrum, sem alterá-lo ou ferir seus valores, de tal forma a amplificar os benefícios do Scrum, que são maiores quando complementados por engenharia aprimorada ou revisada, gerenciamento de produtos, pessoas e práticas organizacionais.

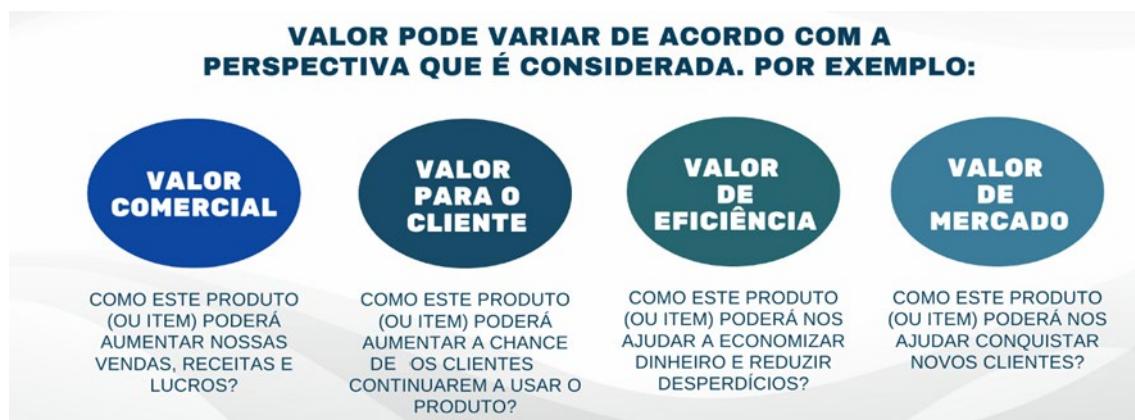


4.3. O que é Produto e o que é Valor?

Um produto é um veículo para entregar valor. Tem um limite claro, stakeholders conhecidos, utilizadores ou clientes bem definidos. Um produto pode ser um serviço, um produto físico ou algo mais abstrato.



E valor, o que significa? Podemos dizer que valor é aquilo que é importante para os stakeholders. Veja a ilustração a seguir.



4.4. Os três compromissos do Scrum

Compromisso é um conceito adotado pelo Scrum para aumentar a transparência e o foco dos artefatos do Scrum. Para cada artefato oficial, há um compromisso atrelado. Desta forma, temos o seguinte:

Artefato	Compromisso
Backlog do Produto	Meta do Produto
Backlog da Sprint	Meta da Sprint
Incremento do Produto	Definição de Pronto

4.5. Compromisso 1: A meta do Produto

A meta do produto descreve um estado futuro do produto que pode servir de meta para o time Scrum delineiar o seu planeamento. A meta do produto está no Backlog do Produto. O resto do Backlog do Produto emerge para definir "o que" a meta do produto irá cumprir. Esta, por sua vez, é o objetivo a longo prazo para o Time Scrum, o qual deve cumprir (ou abandonar) um objetivo antes de assumir o próximo.

Exemplo: Reduzir o tempo total de processamento de pedidos de compra de 3 para 1 dia.

4.6. Compromisso 2: A meta da Sprint

A meta da sprint é o único objetivo para a Sprint. É a razão de existir da Sprint. Responde à pergunta: "porque estamos trabalhando nesta sprint?".

Embora a meta da sprint seja um compromisso dos Desenvolvedores, proporciona flexibilidade em termos do trabalho exato necessário para alcançar a meta. A meta da sprint também cria coerência e foco, encorajando a Time Scrum a trabalhar em conjunto e não em iniciativas separadas.

A meta da sprint é criada durante o Planejamento da Sprint e depois adicionada ao Backlog da Sprint. Durante a Sprint, os Desenvolvedores mantêm esta meta em mente.

4.7. Compromisso 3: A definição de Pronto

A *Definition of Done* ou Definição de Pronto é uma descrição formal do estado do Incremento quando este cumpre as medidas de qualidade exigidas para o produto. **No momento em que um item do Backlog do Produto satisfaz a Definition of Done, nasce um Incremento.**

A *Definition of Done* cria transparência ao proporcionar a todos uma compreensão partilhada do trabalho que foi concluído como parte do Incremento. Se um item de Backlog do Produto não cumpre a Definição de Pronto, não pode ser lançado ou mesmo apresentado na Sprint Review. Em vez disso, regressa ao Backlog do Produto para consideração futura.

DEFINIÇÃO DE PRONTO

DoD - Exemplo 1

- ✓ Código completo de acordo com os padrões de codificação.
- ✓ Código coberto por testes unitários.
- ✓ Código revisado por pares (peer review).
- ✓ Build automático.
- ✓ Não há tarefas pendentes para a funcionalidade.
- ✓ Sem defeitos conhecidos abertos.
- ✓ Todos os testes de aceitação executados.
- ✓ Manual de usuário atualizado.
- ✓ Testes de integração realizados.
- ✓ Testes de regressão realizados.

DoD - Exemplo 2

- ✓ Código desenvolvido e testado.
- ✓ Testes unitários com cobertura de 80% (novos desenvolvimentos).
- ✓ Testes unitários executados e sem falhas.
- ✓ Bugs corrigidos e re-testados.
- ✓ Funcionalidades revisadas pelo Dono do Produto.
- ✓ Implantado em ambiente de homologação.
- ✓ Scripts ajustados para *deploy*.
- ✓ Release notes atualizadas.

FONTE: KELVIN WEISS

5. Pilares e Valores do Scrum

5.1. Para que servem os pilares do Scrum?

Para trabalhos complexos, o controle empírico de processos – ou empirismo – é a abordagem mais adequada, na qual há mais coisas que não sabemos do que que sabemos.

O controle empírico de processos significa trabalhar baseado em fatos, experiência e evidências.

O Scrum funciona por meio de um processo empírico cujo mecanismo está baseado em selecionar um pouco de trabalho, realizá-lo e, de acordo com o resultado, mudar (ou não) a direção.

Desta maneira, informações são obtidas pela observação e não pela previsão. Logo, ao final de cada iteração, pode-se **mudar** a direção dos trabalhos (agilidade).

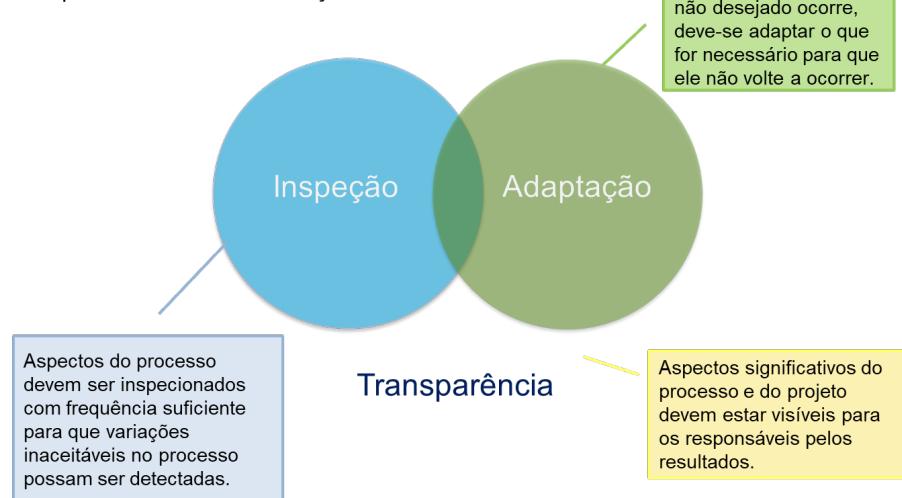


Para que um processo empírico funcione, são necessários três pilares:

- Inspeção
- Adaptação
- Transparência

5.2. Os três pilares do Scrum

Três pilares sustentam a utilização do Scrum:



Transparência

- Exige que aspectos significativos do processo e do projeto estejam visíveis para os responsáveis pelos resultados, sempre!
- As informações devem ser passadas de forma clara e precisa, possibilitando que todos tenham o mesmo entendimento.
- É um pilar que deve ser praticado por todos os papéis e envolvidos no projeto sem exceção!
- Ter transparência é falar a mesma coisa independente de quem fez a pergunta.

Inspeção

- Significa que os processos, práticas e atividades devem ser analisados com frequência suficiente para que as variações inaceitáveis sejam detectadas o mais cedo possível.
- Evita que o cliente receba um produto com qualidade inadequada.

Adaptação

- Significa que sempre que um evento não desejado ocorrer, deve-se adaptar o que for necessário no processo para evitar a sua recorrência.
- Inspeção e adaptação costumam ocorrer juntas.
- O time é que deve realizar.

Adaptação depende de inspeção e inspeção depende de transparência.

5.3. Como o Scrum dá vida aos três pilares

Os eventos de Planejamento da Sprint, Reunião Diária, Revisão da Sprint e Retrospectiva da Sprint são na verdade momentos formais de inspeção de adaptação. Veja na ilustração a seguir.

EVENTOS X PILARES		
Evento	Inspeção	Adaptação
Planejamento da Sprint	▪ Backlog do Produto	▪ Backlog da Sprint
Reunião diária	▪ Progresso da Sprint ▪ (Sprint Goal)	▪ Backlog da Sprint
Revisão da Sprint	▪ Incremento do produto ▪ Time & colaboração	▪ Backlog do Produto
Retrospectiva da Sprint	▪ Tecnologia & engenharia ▪ Processo	▪ Definição de pronto ▪ Ações de melhoria

FONTE: SCRUM.ORG

5.4. Os valores do Scrum

O Scrum também define 5 valores que, quando incorporados, seguidos e vividos pela equipe Scrum, fazem com que os pilares transparência, inspeção e adaptação se tornem uma realidade. Sem seguir os

valores, dificilmente uma equipe pode tirar o máximo de proveito do uso do Scrum. O sucesso no uso do Scrum depende de as pessoas se tornarem mais proficientes nestes cinco valores.

Os cinco valores do Scrum são:



Estes valores dão orientação ao Time Scrum no que diz respeito ao seu trabalho, ações e comportamento. As decisões tomadas, os passos dados e a forma como o Scrum é utilizado devem reforçá-los e não os reduzir ou minar.

Os membros do Time Scrum aprendem e exploram os valores enquanto trabalham com os eventos e os artefatos do Scrum. Quando estes valores são encarnados pelo Time e pelas pessoas com quem eles trabalham, os pilares empíricos da transparência, da inspeção e da adaptação ganham vida construindo confiança.

Coragem

- Mostramos coragem quando não construímos coisas que ninguém quer. Coragem em admitir que requisitos nunca vão ser perfeitos e que nenhum plano pode captar a realidade e a complexidade.
- Coragem de considerar a mudança como uma fonte de inspiração e inovação.
- Coragem para não entregar software que não esteja pronto.
- Coragem em compartilhar toda a informação possível que possa ajudar o time e a organização.
- Coragem em admitir que ninguém é perfeito.
- Coragem para compartilhar riscos e benefícios.
- Coragem para promover o Scrum e o empirismo a fim de lidar com a complexidade.
- Mostramos coragem ao apoiar os valores do Scrum.

Foco

- Uma abordagem iterativa e incremental como o Scrum e o conceito de *timeboxing* permitem que tenhamos foco.
- Nós focamos no que é mais importante agora sem perder muito tempo com aquilo que talvez um dia possa ser importante
- Nós não adicionamos funcionalidades ao software até que estas sejam realmente necessárias.
- Nós focamos no trabalho para fazer as coisas da maneira mais simples e objetiva possível.

Comprometimento

- Comprometimento não é só sobre alcançar um resultado, cumprir uma meta. É como você se comporta durante os trabalhos para tentar atingir os resultados.
- O comprometimento é sobre dedicação e esforço empregado às ações. É sobre fazer o seu melhor sempre!
- Nós temos comprometimento com o time, com a qualidade, com a colaboração.
- Temos comprometimento com o aprendizado e em fazer o melhor que podemos todos os dias.
- Temos comprometimento com a melhoria contínua e com a excelência técnica.
- Temos comprometimento com os valores do Scrum e os princípios ágeis.

Respeito

- Mostramos respeito pelas pessoas e por suas experiências profissionais e pessoais.
- Nós respeitamos as habilidades, experiências e conhecimentos uns dos outros.
- Nós respeitamos a diversidade (que nos torna mais fortes).
- Mostramos respeito aos nossos *stakeholders* quando não construímos funcionalidades que ninguém vai usar.
- Mostramos respeito por não desperdiçar dinheiro em coisas que não são valiosas ou talvez nunca serão implementadas ou usadas.
- Mostramos respeito aos usuários corrigindo rapidamente os erros.
- Nós respeitamos o framework Scrum.

Abertura

- O empirismo do Scrum requer transparência e abertura.
- Estamos abertos sobre o nosso trabalho, o nosso progresso, a nossa aprendizagem e os nossos problemas.
- Mas também estamos abertos para as pessoas, o que implica reconhecer que as pessoas são pessoas e não recursos, robôs ou peças substituíveis de uma máquina.
- Estamos abertos para colaborar em todas as disciplinas e competências.
- Estamos abertos para colaborar com as partes interessadas e o ambiente em geral.
- Abertos para compartilhar feedback e aprender uns com os outros.

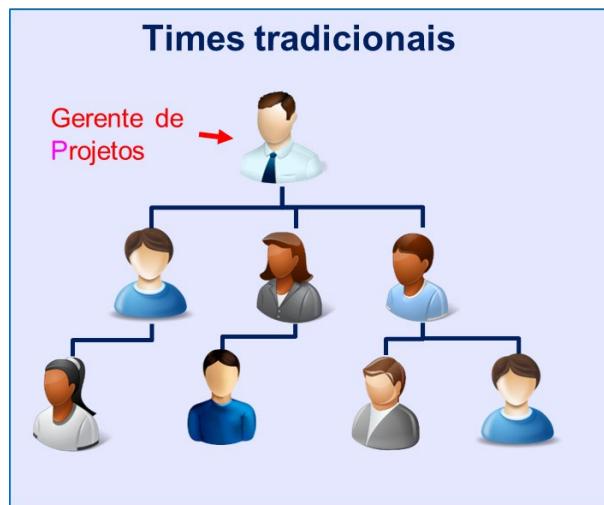
6. Pessoas & Time Scrum

6.1. Os problemas com equipes tradicionais

Ao longo dos anos, as organizações foram divididas em departamentos independentes, cada qual com a sua especialidade, objetivo e prioridade: Marketing, Operações, Finanças, Tecnologia da Informação (T.I.), Recursos Humanos (R.H.), Administração etc. Dentro de cada departamento, as pessoas foram divididas em unidades ainda mais especializadas. Por exemplo: Finanças tem a unidade (time) de Contas a Pagar, Aquisições, Controle de Orçamento, entre outras.

A ideia por trás dessa divisão era que a soma das partes gera bom resultado para o todo. Infelizmente isso é uma falácia! A divisão cria silos de conhecimento e cada unidade constrói a sua própria agenda.

Trabalhar em silos pode impactar seriamente o seu negócio. Eles causam conflitos internos e podem despertar a falta de confiança dentro da empresa, o que resulta em ineficiência e redundâncias no interior dos departamentos. Em termos práticos, as equipes deixam de se falar, os profissionais param de receber feedback e as falhas na comunicação podem desencaminhar projetos. Muito rapidamente, sua estrutura organizacional começa a minar inovações, e pequenas rachaduras na fundação começam a parecer intransponíveis.



Esse definitivamente não é o jeito com que o Scrum trabalha.

6.2. O Time Scrum

A unidade fundamental do Scrum é uma pequena equipe de pessoas, um Time Scrum que é composto por um **Scrum Master**, um **Dono de Produto** e **Desenvolvedores**.



Não existe qualquer outro papel

Dentro de um Time Scrum, não existem subequipes ou hierarquias. É uma unidade coesa de profissionais, focada num objetivo de cada vez, a Meta do Produto.

O Time Scrum é responsável por todas as atividades relacionadas aos produtos, desde a colaboração dos stakeholders, verificação, manutenção, operação, experimentação, investigação e desenvolvimento a tudo o mais que possa ser necessário.

Ele é estruturado e capacitado pela organização para gerir o seu próprio trabalho. Trabalhar em Sprints a um ritmo sustentável melhora o foco e a consistência do Scrum Team.



Os Times Scrum são multifuncionais, o que significa que os membros possuem todas as habilidades necessárias para criar valor a cada Sprint. Eles também são autogerenciáveis, o que quer dizer que decidem internamente quem faz o que, quando e como.

6.3. O que é ser autogerenciado?

Um Time Scrum é autogerenciado, o que significa que os seus membros decidem internamente quem faz o que, quando e como. O foco está em entregar a Meta do Produto e não simplesmente itens de trabalho ou tarefas.

O autogerenciamento gera criatividade, habilidade para resolver conflitos e sentimento de responsabilidade.

AUTOGERENCIAMENTO É UM MODELO ONDE NÃO HÁ UMA AUTORIDADE CENTRAL OU ELEMENTO EXTERNO QUE DETERMINE O QUE, QUANDO E COMO AS COISAS DEVEM SER FEITAS.

6.4. O tamanho do Time Scrum

O Time Scrum é pequeno o suficiente para permanecer ágil e grande o suficiente para concluir um trabalho significativo dentro de uma Sprint. Isso significa que TIPICAMENTE um time Scrum tem 10 ou menos pessoas. Mas pode haver mais também. Não é uma regra, apenas uma recomendação.

Em geral, descobrimos que times menores se comunicam melhor e são mais produtivos. Se eles se tornarem muito grandes, devem considerar a reorganização em vários times menores e coesos, cada um focado no mesmo produto. Portanto, eles devem compartilhar a mesma meta do produto, Backlog do Produto e Dono de Produto.

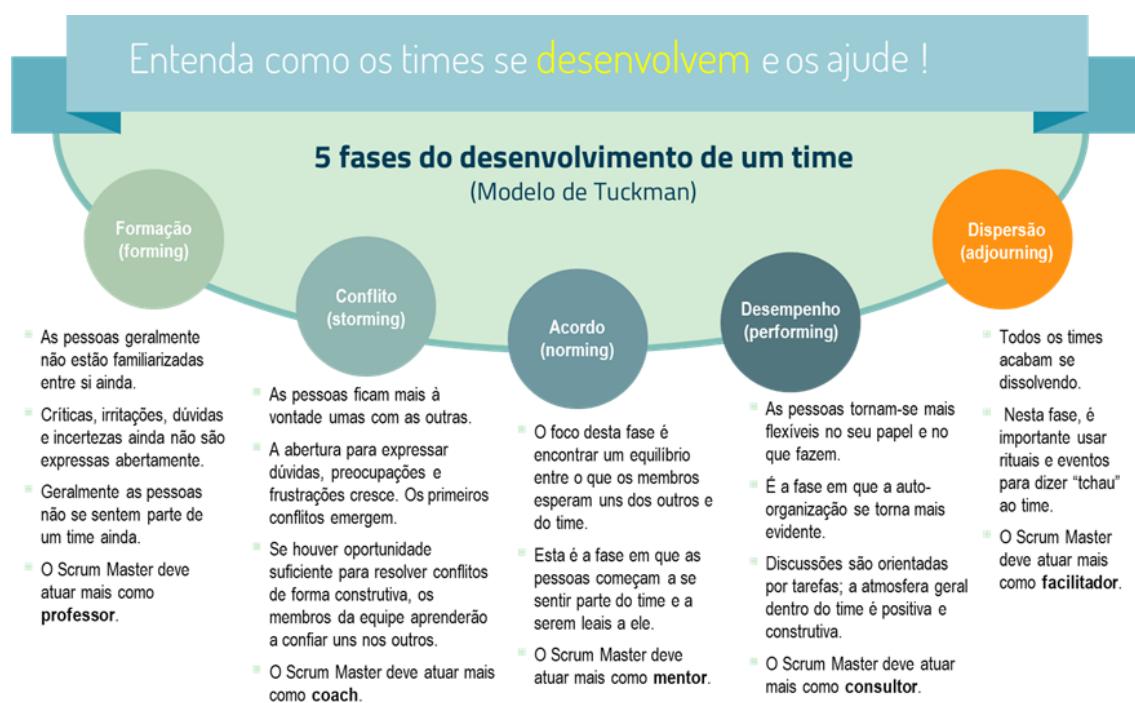
6.5. O Time Scrum é multifuncional

Os Times Scrum são multifuncionais, o que significa que os membros têm todas as competências necessárias para criar valor em cada Sprint. É importante reforçar que é o time que é multifuncional e não os membros dele.

Logo, cada integrante do time Scrum poderá ter um conjunto de habilidades específicas, enquanto permanece responsável pelo todo. Obviamente quanto maior for o conjunto de habilidades individuais, melhor será para a organização do trabalho.

6.6. O modelo de Tuckman e seus impactos

O modelo de **Tuckman** descreve os estágios para o desenvolvimento de times. O modelo assume que os grupos possuem, de forma geral, comportamentos padronizados, que passam por cinco etapas: formação, tormenta, normalização, desempenho e interrupção.



A definição destes estágios foi fruto de uma extensa revisão de literatura de Bruce Tuckman. O autor analisou diversos artigos científicos da época os quais tratavam da formação de grupos e sintetizou as conclusões de cada autor analisado em cinco fases gerais dentro do processo de formação de um time.

Mudanças na composição do time, mesmo que sejam de apenas uma pessoa (adicionada ou removida), geralmente retornam à "fase de Acordo".

6.7. Times remotos e o Scrum

É perfeitamente possível utilizar o Scrum mesmo em times 100% remotos. É um mito achar que apenas times colocalizados (no mesmo espaço físico) podem usar o Scrum.

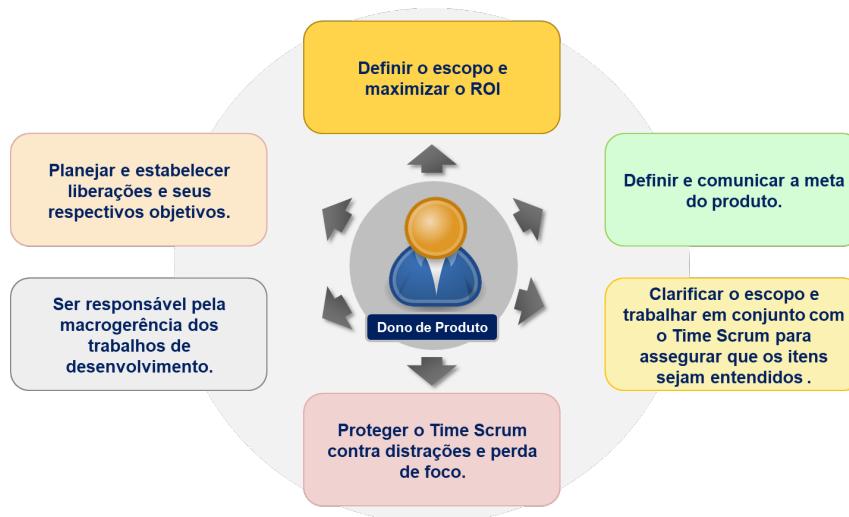
6.8. O Dono do Produto

O Dono de Produto é responsável por maximizar o valor do produto resultante do trabalho do Scrum Team. A forma como isso é feito pode variar amplamente entre organizações, Times Scrum e indivíduos.

O Dono de Produto também é responsável pelo gerenciamento eficaz do Backlog do Produto, o que inclui:

- Desenvolver e comunicar explicitamente a meta do produto;
- Criar e comunicar claramente os itens do Backlog do Produto;
- Ordenar os itens do Backlog do Produto;
- Garantir que o Backlog do Produto seja transparente, visível e compreensível.

O Dono de Produto pode fazer o trabalho acima ou delegar a responsabilidade a outros. Independentemente disso, o Dono de Produto ainda é o responsável.



6.9. A autoridade do Dono de Produto

Para que o Dono de Produto tenha sucesso, toda a organização deve respeitar suas decisões, as quais são visíveis no conteúdo e na ordem do Backlog do Produto e por meio do incremento inspecionável na revisão da sprint.

O Dono de Produto é uma pessoa, não um comitê. Ele pode representar as necessidades de muitos stakeholders no Backlog do Produto. Aqueles que desejam alterar o Backlog do Produto podem fazê-lo tentando convencer o Dono de Produto.

6.10. O perfil de um Dono de Produto

- É o representante do cliente, dos usuários ou de qualquer um com interesse no produto que será desenvolvido.
- É essencial que ele conheça o negócio que vai resultar em um produto bem como as reais necessidades dos usuários.
- Bem-informado sobre o cliente e o produto.
- Capacitado e capaz de tomar decisões claras e rápidas para manter o Time Scrum avançando.

- Disponível para os Desenvolvedores a fim de esclarecer metas e produção desejada.
- Responsável pelo sucesso comercial do produto.

6.11. Os Desenvolvedores

Desenvolvedores são as pessoas do Time Scrum que estão comprometidas em criar qualquer aspecto de um Incremento utilizável a cada Sprint.

As habilidades específicas necessárias aos Desenvolvedores geralmente são amplas e variam de acordo com o domínio de trabalho. No entanto, eles são sempre responsáveis por:

- Criar um plano para a Sprint, o Sprint Backlog;
- Introduzir gradualmente qualidade aderindo a uma Definition of Done;
- Adaptar seu plano a cada dia em direção à meta da Sprint; e,
- Responsabilizar-se mutuamente com os demais profissionais.

6.12. O Scrum Master

O Scrum Master é responsável por estabelecer o Scrum conforme definido no Guia do Scrum. Ele faz isso ajudando todos a entender a teoria e a prática do Scrum, tanto no Scrum Team quanto na organização.

Trata-se do responsável pela eficácia do Scrum Team. Ele faz isso possibilitando que o Time Scrum melhore suas práticas dentro do framework Scrum.

O Scrum Master não é chefe dos Desenvolvedores ou Dono de Produto. Ele também não é um Gerente de Projetos. É um **verdadeiro líder que serve**:



O Scrum Master melhora o dia a dia de todo o Time Scrum, à medida que:

- Estimula a criatividade e o fortalecimento do trabalho em equipe.
- Serve como um mediador entre as diferentes partes.
- Remove as barreiras entre Desenvolvedores e o Dono de Produto para que as duas partes possam interagir diretamente.
- Combate o estilo comando e controle.
- Faz o máximo para ensinar ao Time Scrum o que é e como ser um time autogerenciado.

- Garante o uso correto do Scrum. Ele deve defender e ser ponto de referência para aplicação dos valores e práticas do Scrum por todos os envolvidos.
- Garante que as práticas do Scrum estão sendo seguidas da maneira correta.
- Dissemina a cultura do Scrum na organização.

O Scrum Master é uma posição de gestão, mas não de gestão de pessoas e sim de gestão do processo. Ele é o líder e guardião do processo Scrum.

6.13. A quem o Scrum Master serve?

Scrum Masters são verdadeiros líderes que servem ao Scrum Team e à organização como um todo.

O Scrum Master serve ao Time Scrum de várias maneiras, incluindo:

- Treinar os membros do time em autogerenciamento e cross-funcionalidade;
- Ajudar o Scrum Team a se concentrar na criação de incrementos de alto valor que atendam à Definition of Done;
- Provocar a remoção de impedimentos ao progresso do Scrum Team;
- Garantir que todos os eventos Scrum ocorram e sejam positivos, produtivos e mantidos dentro do Timebox.

O Scrum Master serve ao Dono de Produto de várias maneiras, incluindo:

- Ajudar a encontrar técnicas para a definição eficaz de meta do Produto e gerenciamento do Product Backlog;
- Ajudar o Scrum Team a entender a necessidade de itens do Product Backlog claros e concisos;
- Ajudar a estabelecer o planejamento empírico do produto para um ambiente complexo;
- Facilitar a colaboração dos stakeholders, conforme solicitado ou necessário.

O Scrum Master serve à Organização de várias maneiras, incluindo:

- Liderar, treinar e orientar a organização na adoção do Scrum;
- Planejar e aconselhar implementações de Scrum dentro da organização;
- Ajudar os funcionários e os stakeholders a compreenderem e aplicarem uma abordagem empírica para trabalhos complexos;
- Remover barreiras entre stakeholders e Scrum Teams.

6.14. Arquitetura do Time Scrum

Não há nada no Scrum que impeça uma mesma pessoa de assumir mais de uma responsabilidade, como por exemplo, ser Dono de Produto e Scrum Master ao mesmo tempo ou ser Scrum Master e Desenvolvedor. No entanto saiba que isso **pode** causar impactos negativos, como falta de foco ou conflito de interesse. Mas novamente: não é proibido.

Da mesma forma, um mesmo desenvolvedor pode fazer parte de mais de um time Scrum, mas saiba que possivelmente haverá perda de produtividade em função do foco dividido.

6.15. E o Gerente de Projetos?

Agora que nós analisamos todos os papéis do Scrum, você pode se perguntar: quem é o gerente de projeto? A resposta é simples: **não existe tal papel no Scrum**.

Algumas pessoas consideram o Scrum Master como equivalente a um gerente de projeto tradicional, mas isso não é verdade porque as responsabilidades deste são muito diferentes das daquele. Scrum Masters não são responsáveis pelo planejamento, por exemplo.

Outros podem considerar o Dono do Produto como equivalente a um gerente de projeto tradicional, o que também não é correto. Ainda que o Dono de Produto seja responsável por partes do planejamento e do monitoramento do projeto, isso também é parcialmente feito pelos Desenvolvedores, os quais também fazem a gestão de tarefas.

Se não existe o papel de gerente de projeto, quem gerencia o projeto? Parte das atribuições de um típico gerente de projeto agora são repassadas para outros papéis.

Responsabilidade	Dono do Produto	Desenvolvedores	Scrum Master
Escopo	✓ (produto)	✓ (Sprint)	✗
Tempo	✓ (produto)	✓ (Sprint)	✗
Custo	✓	✗	✗
Comunicação	✓	✓ (Sprint)	✗
Risco	✓	✓	✓
Qualidade	✓ (escopo)	✓ (testes)	✓ (processo)

6.16. E os gerentes funcionais?

Cada Time Scrum geralmente funciona dentro de um ecossistema organizacional maior. Embora o Scrum não tenha funções de gestão, como gerentes de departamento ou projetos, há um entendimento sobre a importância de "gestores organizacionais".

Esses gerentes estabelecem e se responsabilizam por estratégias maiores, definem as unidades operacionais, estruturam times autogerenciados e ajudam a resolver impedimentos organizacionais para agilidade.

Se os problemas estão fora do contexto do Time Scrum, tais como contratação de pessoas, demissão, remuneração e outros aspectos legais, eles devem ser tratados pela gestão da empresa.

7. Gerenciamento Ágil do Escopo

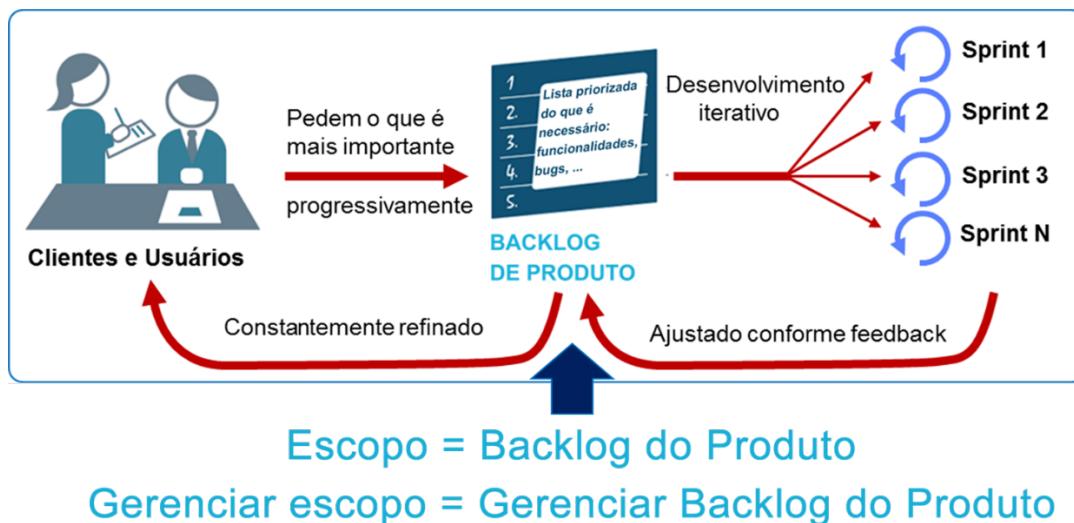
7.1. Gerenciando o escopo de forma ágil

Como funciona o gerenciamento de escopo no Scrum? Bem, os clientes e usuários pedem o que é mais importante e de forma progressiva. O Dono do Produto então registra essas solicitações no Backlog do Produto. Uma vez que se tenha um backlog inicial, começa o desenvolvimento de forma iterativa, realizando a Sprint 1, a 2, a 3 e assim por diante.

Perceba que um projeto pode ser constituído de um conjunto de sprints – 5 sprints, 10 sprints, 15 sprints – quantas forem necessárias e esse número pode ser pré-definido ou não. Podemos iniciar um projeto sabendo que iremos executar 10 sprints, por exemplo, ou então ir efetuando-as, uma após a outra até que o cliente ou patrocinador encerre o projeto. Assim as Sprints são realizadas várias vezes, até que:

- O produto esteja pronto porque:
 - Todos os itens do Backlog do Produto estão prontos.
 - Ou porque o cliente avaliou que a mais recente iteração é suficiente e não há justificativa para gastar mais tempo e dinheiro adicionando mais recursos.
- Ou então o desenvolvimento do produto é encerrado por algum outro motivo (por exemplo, não se justifica mais).

Observe na imagem abaixo que ao final de cada sprint o backlog do produto é ajustado conforme o feedback dado pelo cliente. Dessa forma, este backlog é constantemente refinado.



7.2. O Backlog do Produto

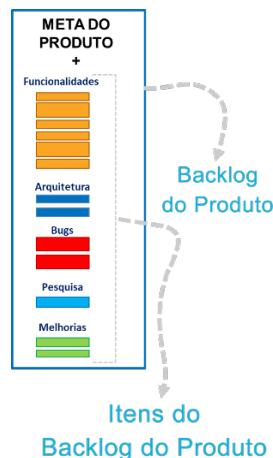
O Backlog do Produto mostra todo o trabalho que está previsto para o desenvolvimento do projeto. É uma lista ordenada de tudo aquilo que pode ser necessário no produto.

Qualquer intervenção necessária para ou no produto deve estar no Backlog do Produto e o que estiver fora dele não será desenvolvido. O Product Backlog traz à tona todo o trabalho, desenvolvimento, premissas e restrições com os quais um time tem que lidar a fim de criar um produto pronto e potencialmente liberável.

O Backlog do Produto pode compreender itens funcionais e não funcionais, melhorias, correções, ideias, atualizações e outros requisitos.

Observe a imagem a seguir: O Backlog do Produto é formado por diversos itens que são chamados de “itens do Backlog do Produto”. Eles podem ser de diversas naturezas, desde que agreguem valor ao

produto ou a clientes e usuário e precisam ser valiosos. Qualquer coisa que não seja valiosa para o produto não deveria ser feita.



Se alguém quer saber quais itens estão identificados e planejados para serem desenvolvidos, deve olhar para o Backlog do Produto. Ele é a única fonte de requisitos para que as alterações sejam feitas no produto. O Time Scrum não enxerga outra origem de trabalho e o Dono de Produto não deve pedir coisas que não estejam no backlog.

EXEMPLO BACKLOG DE PRODUTO

ITENS DE BACKLOG DE PRODUTO

Descrição (Description)	Prioridade (Order)	Valor (Value)	Estimativa (Estimate)
Como um gerente de hotel gostaria de ter uma listagem de todos os quartos do hotel para poder o verificar o status de cada um	500	1 - Alto	5
Como um gerente de hotel gostaria de saber todos os quartos que foram alugados no mês para saber quais tipos de quartos foram mais alugados	460	2 - Médio	8
Como um cliente gostaria de consultar os preços dos quartos disponíveis para que eu possa alugá-los	450	1 - Alto	13
Como um cliente gostaria de visualizar os quartos disponíveis para poder realizar uma reserva	440	1 - Alto	8
Como um cliente gostaria de poder cancelar uma reserva já realizada para que eu não pague taxas de no-show	400	1 - Alto	13
Como um gerente de hotel gostaria de apurar os valores consolidados mês a mês para monitorar o desempenho	230	2 - Médio	8
Como um gerente de hotel			

COM ITENS REFERENTE AO DESENVOLVIMENTO DE SOFTWARE

7.3. A meta do produto

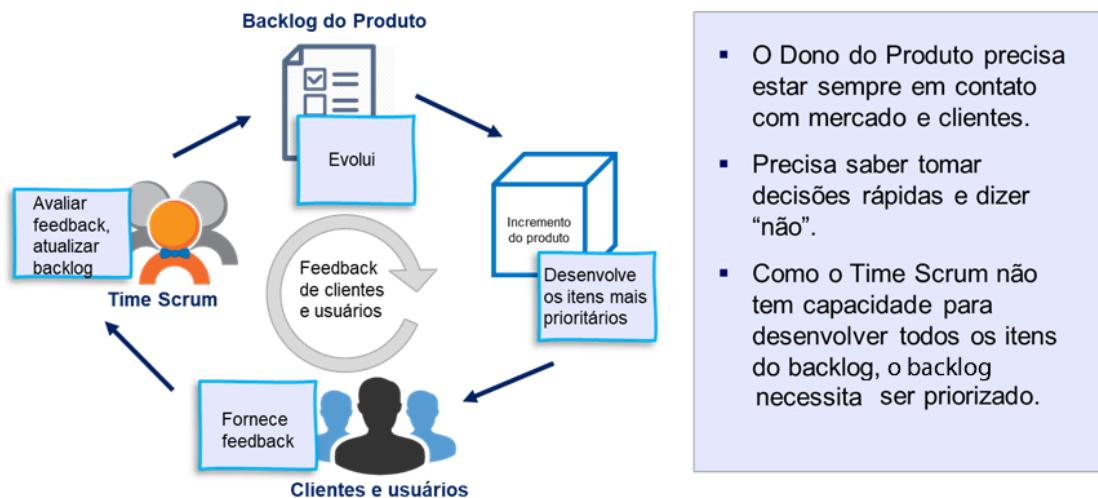
A Meta do Produto descreve um estado futuro do produto que pode servir como um alvo para o Time Scrum planejar. Ela está no Product Backlog, que emerge para definir “o que” cumprirá a Meta do Produto.

A Meta do Produto serve para aumentar o foco e a transparência. Ela é o objetivo de longo prazo para o Scrum Team, que deve cumprir (ou abandonar) um objetivo antes de assumir o próximo.



7.4. O Backlog do Produto é emergente

O Backlog do Produto evolui ao longo do processo, conforme o produto se desenvolve. Isso significa que ele é dinâmico e está constantemente mudando para identificar o que o produto precisa ter para ser adequado, competitivo e útil para clientes e usuários. O Backlog do Produto é priorizado novamente ao menos uma vez no início de cada Sprint.



7.5. Entendendo os PBIs

Um item de backlog de produto é algo que precisa ser feito no produto ou para o produto. Pode ser uma funcionalidade, uma característica, uma correção de erro, uma ideia, uma necessidade ou qualquer outra coisa que quando realizada agregue valor para clientes, usuários ou stakeholders.

A forma como um PBI (Product Backlog Item ou Item de Backlog de Produto) é descrito vai variar de ambiente para ambiente, de negócio para negócio. Tipicamente se incluem as seguintes informações:

Descrição

É o que precisa ser feito, o que necessita ser desenvolvido pelos Desenvolvedores. Sempre que possível, deve ser escrita na linguagem de negócio, ou seja, o item é orientado pelas necessidades de negócio e não pelas de tecnologia. Evitar entrar em detalhes técnicos de como fazer o item. Por exemplo (usando o mundo de desenvolvimento de software): a emissão de notas fiscais deve ser desenvolvida com uso de threads. Isso é uma decisão técnica e não uma necessidade negócio. Esse mesmo item poderia ser descrito como: a emissão de notas fiscais deve ser escalável, rodar em servidores distribuídos e com alta performance. Essa é uma necessidade de negócio. Depois a equipe técnica avalia como implementar isso tecnicamente.

Valor

É o valor de cada item na perspectiva do negócio. É o quanto valioso é o item para clientes e usuários. Falando de outra forma, é o benefício daquele item na perspectiva do cliente. Por exemplo: este item tem um benefício muito alto, porque ele vai permitir que os usuários façam o seu trabalho na metade do tempo.

Esforço ou outra forma de dimensionamento de tempo e/ou custo

É o custo para se fazer cada item. Normalmente, utilizam-se pontos de complexidade (ou user point) para estimar os itens. Por exemplo: fazer determinado item custa 8 pontos. Este outro custa 13 e assim por diante. É, na verdade, uma estimativa de tamanho de cada item. Pense no m² da construção civil. Um m² pode levar 5 ou 10 horas para ser feito, vai depender de diversos fatores. O mesmo ocorre com os pontos de complexidade. O que queremos aqui é criar uma base para comparação, poder comparar dois itens diferentes. Lembre-se de que, durante a reunião de planejamento da sprint, os Desenvolvedores estimam os itens e depois os selecionam para a sprint de acordo com sua capacidade produtiva, que também é medida por meio de pontos. Mas isso é apenas uma das possíveis formas de se fazer isso, apenas para que você entenda o contexto como um todo. São os Desenvolvedores que estimam cada item. Então o Dono de Produto avalia o benefício e os Desenvolvedores estabelecem o custo.

Ordem

É a ordem em que cada um dos itens será desenvolvido. Itens com prioridade maior recebem mais atenção do Dono de Produto e dos Desenvolvedores. Lembre-se de que no Scrum as coisas vão sendo feitas de forma iterativa e incremental. Então, quanto mais próximo do desenvolvimento do item, maior atenção ele recebe. Podem ser utilizadas várias técnicas para priorizar um item. Mais tarde falaremos sobre elas. É através da prioridade que o Dono de Produto maximiza o ROI e o trabalho dos Desenvolvedores. Aquilo que ele prioriza para ser feito primeiro precisa ter o maior valor possível para clientes e usuários.

O PBI pode ter outras informações? Pode. Essas quatro são as mais básicas, mas você pode ter uma coluna para agrupar os itens por assunto, outra para registrar a data em que o item foi cadastrado se isso for importante para você etc. É possível customizar do jeito que quiser o seu Backlog do Produto.

7.6. Ordenando o Backlog de Produto

Uma das atividades mais importantes, se não a mais importante, pela qual o Dono do Produto é responsável é ordenar os itens do Backlog do Produto. Se eles não tiverem uma ordem adequada, o Scrum como um todo perde grande parte do seu valor porque o time Scrum começará a desenvolver coisas de menor importância para o cliente e usuários ou que vão gerar menos valor.

O Dono do produto tem autoridade para priorizar os itens e com isso ele ganha automaticamente uma grande responsabilidade nesta etapa do processo.

A regra para priorização do Backlog do Produto é utilizar o valor de negócios de cada item.

$$\text{Valor de Negócio} = \frac{\text{Benefício}}{\text{Custo}}$$

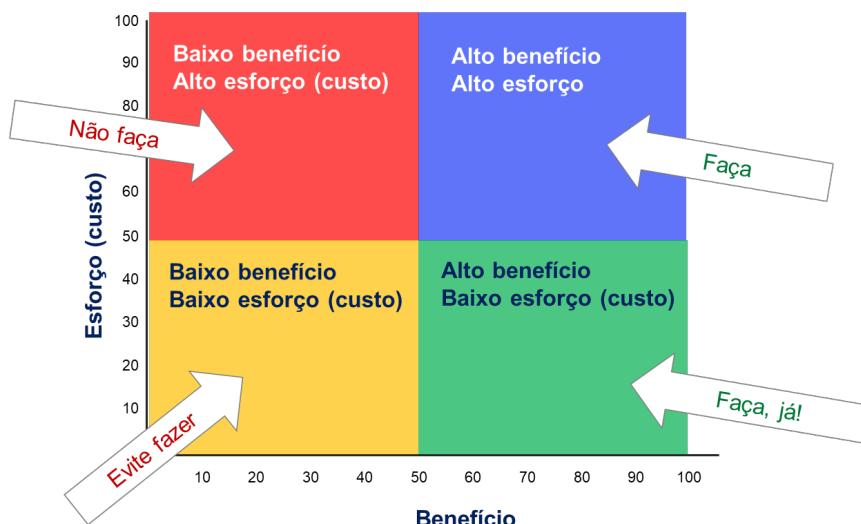
Riscos positivos e/ou negativos estão incluídos como também questões de dependências

Existem diversas técnicas para calcular o valor de negócio de cada item. Não há uma única forma e o Scrum Guide não prescreve nenhuma técnica para isso. O Dono do Produto é livre para escolher o seu método ou uma combinação deles. O importante é garantir o máximo possível de retorno para o negócio e otimizar o trabalho dos Desenvolvedores, entregando o que é mais importante primeiro.

Uma das técnicas que o Dono de Produto precisa saber usar com maestria é a análise de Custo Benefício para decidir o que deve ser feito primeiro. Essa técnica confronta o benefício de cada item do backlog, contra o esforço, ou seja, contra o custo de se fazer o item. Isso gera então uma matriz que ajuda na organização das ideias.

O benefício de cada item deve ser identificado pelo Dono de Produto e o custo, ou seja, o esforço de desenvolvimento, deve ser indicado pelos Desenvolvedores.

Observe que a matriz abaixo nos mostra alguns quadrantes:



Obviamente o que tem um alto benefício e um baixo custo – representado pela área verde – deveria ser a primeira coisa a ser feita. Pense: algo que vai dar um ótimo retorno e é baratinho para fazer torna o item um sonho de consumo. Deve ser nossa prioridade número 1.

Já o que tem baixo benefício e alto custo (representado pela área vermelha) não deveria ser feito. Ou seja, custa caro e não gera retorno, devendo, portanto, ser nossa última prioridade.

A dúvida fica sempre entre qual dos outros dois quadrantes priorizar. Fazer primeiro o que tem alto benefício e alto custo ou o que tem baixo custo e baixo benefício. Na prática, preferimos priorizar aquilo que tem alto benefício e custo elevado, o que é representado pela área em Azul. Isso porque aquilo que tem baixo benefício e baixo custo pode ser comparado ao chuchu: é baratinho para comprar na feira, mas também não tem gosto de nada. Gasta pouquinho, mas o valor também é “inho”.

Então orientamos que primeiro seja feito o que estiver na área verde, depois na azul e em seguida na amarela. Deve-se evitar ao máximo que sejam realizadas as ações que estão na área em vermelho. Esta é uma das coisas que o Dono de Produto mais precisa ter em mente: a relação custo-benefício dos itens.

7.7. O conceito de preparado

Os itens do Backlog do Produto que podem ser realizados pelo Time Scrum em uma Sprint são considerados **preparados** para seleção no evento de Planejamento da Sprint.

O que você leu acima é tudo o que o Scrum Guide descreve sobre o conceito de **preparado**.

No entanto, existe um outro conceito (que NÃO está dentro do Scrum Guide). Ele se chama Definiton of Ready ou Definição de Preparado. A DoR (definition of ready) é aplicada quando se deseja deixar claro quais características um item de backlog de produto precisa ter ANTES de ser discutido em uma reunião de planejamento da sprint.

A DoR é um tanto polêmica porque dependendo de como é usada pode fazer bem ou mal. Então entenda o seguinte: para a prova da Scrum.Org, não existe a DoR. Qualquer questão ou alternativa que mencione este conceito, provavelmente estará errada.

7.8. Refinamento do Backlog do Produto

Refinamento = preparar, arrumar.

Os itens do Backlog do Produto que podem ser realizados pelo Time Scrum em uma Sprint são considerados preparados para seleção no evento de Planejamento da Sprint. Eles geralmente adquirem esse grau de transparência após as atividades de refinamento.

O **Refinamento do Backlog do Produto** é o ato de quebrar e incluir definição adicional aos itens do Backlog do Produto para que eles se tornem menores e mais precisos. Esta é uma atividade contínua de adicionar detalhes como descrição, ordem e tamanho e pode ser realizada a qualquer momento, sempre que houver necessidade. Os atributos geralmente variam de acordo com o domínio de trabalho.

Os Desenvolvedores que farão o trabalho são responsáveis pelo dimensionamento. O Dono do Produto pode influenciá-los, ajudando-os a entender e selecionar trade-offs (trocas de itens).

O Dono do Produto é o responsável por conduzir o refinamento, cujo objetivo é preparar os itens do Backlog do Produto para entrar nas futuras Sprints. Apesar de isso ser uma tarefa do Dono, ele pode contar com o apoio de todo o Time Scrum.

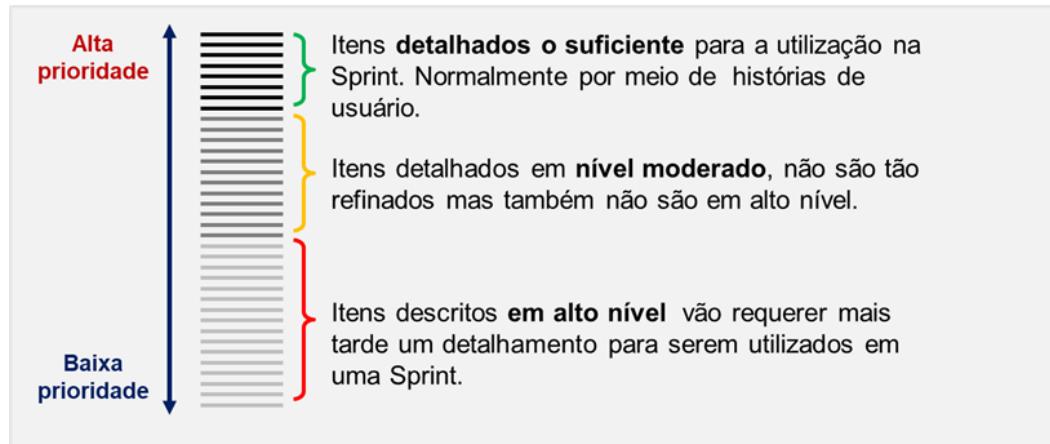
Durante a realização desta atividade, o time avalia a granularidade dos itens do Backlog do Produto e, se necessário, os quebra em itens menores. Uma vez que itens do Backlog do Produto estejam claros o suficiente, os Desenvolvedores podem dimensionar cada um (ex.: estimar a complexidade de cada um).

O refinamento ajuda a nos mostrar que o backlog não é estático. Ao invés disso, ele tem natureza orgânica e vai evoluindo com o passar do tempo. Nem tudo deve receber o mesmo investimento de tempo.

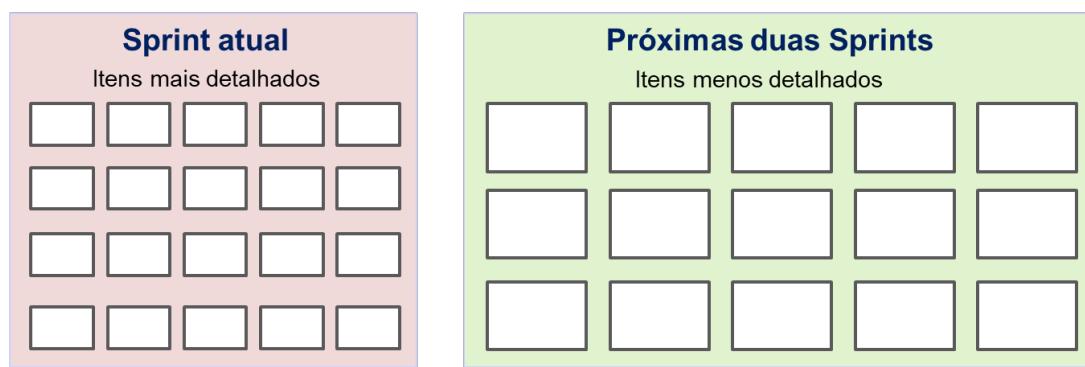
Detalhamento do Backlog do Produto

Um dos pontos mais críticos do Backlog do Produto é como e o quanto os seus itens precisam ser detalhados. O nível de detalhe é definido pelo Time Scrum e pode variar para cada item. Cada item do backlog deve estar PREPARADO para entrar na Sprint.

Um critério que podemos utilizar para o detalhamento é a prioridade de cada item do Backlog do Produto:



Os itens com maior prioridade devem estar mais detalhados e prontos para entrarem em uma Sprint a fim de que os Desenvolvedores iniciem seu trabalho. Os itens do Backlog do Produto devem estar priorizados e estimados por mais ou menos três Sprints.



7.9. A origem da Meta do Produto

O Backlog do Produto nasce juntamente com a visão do produto. Durante o processo de definir a visão do produto, começam a surgir os primeiros itens (funcionalidades) do Backlog do Produto. A partir deste momento, o Dono do Produto começa a registrar os itens e a priorizá-los.

A Meta do Produto se origina da Visão do Produto, que não é obrigatória, mas quando existe ajuda a orientar todo o Time Scrum, promovendo mais alinhamento e autonomia.

EXEMPLO VISÃO DE PRODUTO

Para os peladeiros de plantão
Que tem dificuldade de participar de jogos
O Easy-Bola
É um App Mobile
Que facilita encontrar jogos
Diferentemente de boca-a-boca
O Nosso produto maximiza as chances de acontecimentos das peladas

FONTE: PAULO CAROLI

8. Sprint & DoD & Timebox

8.1. A Definição de Pronto (DoD)

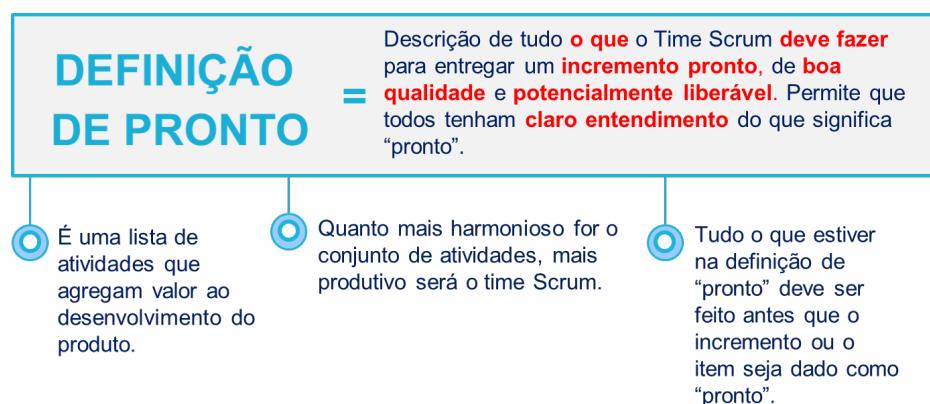
O que é pronto?



Ao **menos** ao final de cada Sprint, um incremento de produto tem que estar **pronto**.



Quando um item do Backlog do Produto ou um incremento é descrito como “pronto”, todos devem entender o que “pronto” significa. Para haver uma maior transparência, deve ser então criada uma “Definição de Pronto (Definition of Done – DoD)”.



Exemplo (para o mundo de desenvolvimento de software):

DoD - Exemplo 1

- ✓ Código completo de acordo com os padrões de codificação.
- ✓ Código coberto por testes unitários.
- ✓ Código revisado por pares (peer review).
- ✓ Build automático.
- ✓ Não há tarefas pendentes para a funcionalidade.
- ✓ Sem defeitos conhecidos abertos.
- ✓ Todos os testes de aceitação executados.
- ✓ Manual de usuário atualizado.
- ✓ Testes de integração realizados.
- ✓ Testes de regressão realizados.

DoD - Exemplo 2

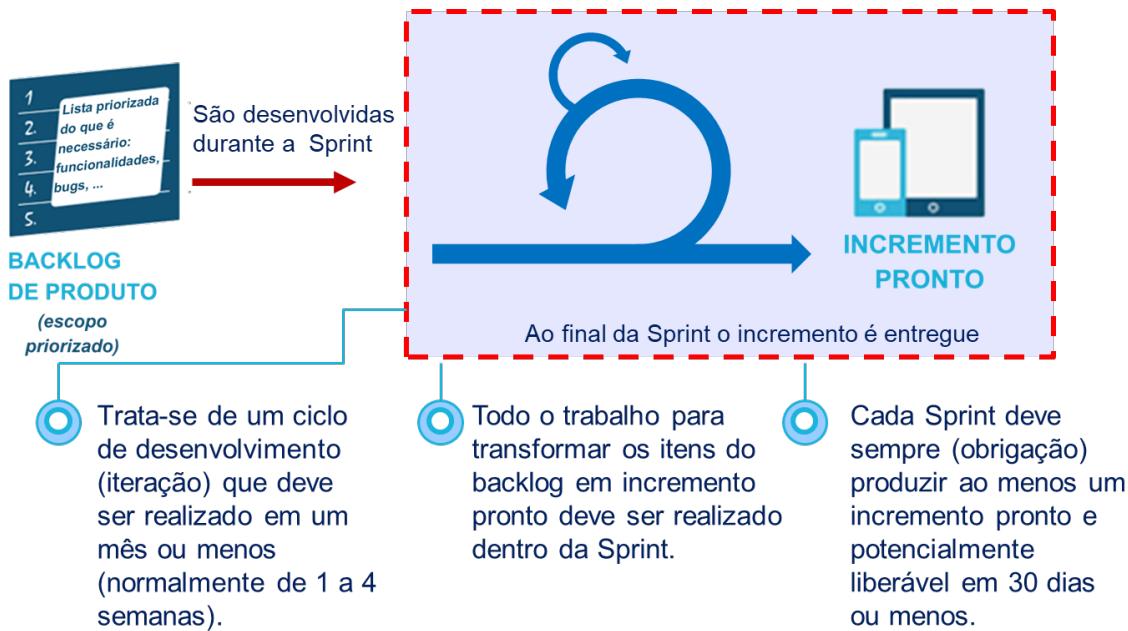
- ✓ Código desenvolvido e testado.
- ✓ Testes unitários com cobertura de 80% (novos desenvolvimentos).
- ✓ Testes unitários executados e sem falhas.
- ✓ Bugs corrigidos e re-testados.
- ✓ Funcionalidades revisadas pelo Dono do Produto.
- ✓ Implantado em ambiente de homologação.
- ✓ Scripts ajustados para *deploy*.
- ✓ Release notes atualizadas.

A Definição de Pronto cria transparência ao fornecer a todos um entendimento compartilhado de qual trabalho foi concluído como parte do Incremento. Se um item do Backlog do Produto não atender à Definição de Pronto, ele não poderá ser liberado ou mesmo apresentado na Revisão da Sprint. Em vez disso, ele retorna ao Backlog do Produto para consideração futura.

Se a Definição de Pronto para um incremento faz parte dos padrões da organização, todos os Times Scrum devem segui-la como mínimo. Se este não for o caso, deve-se criar uma Definição de Pronto adequada para o produto e os Desenvolvedores devem estar em conformidade com ela.

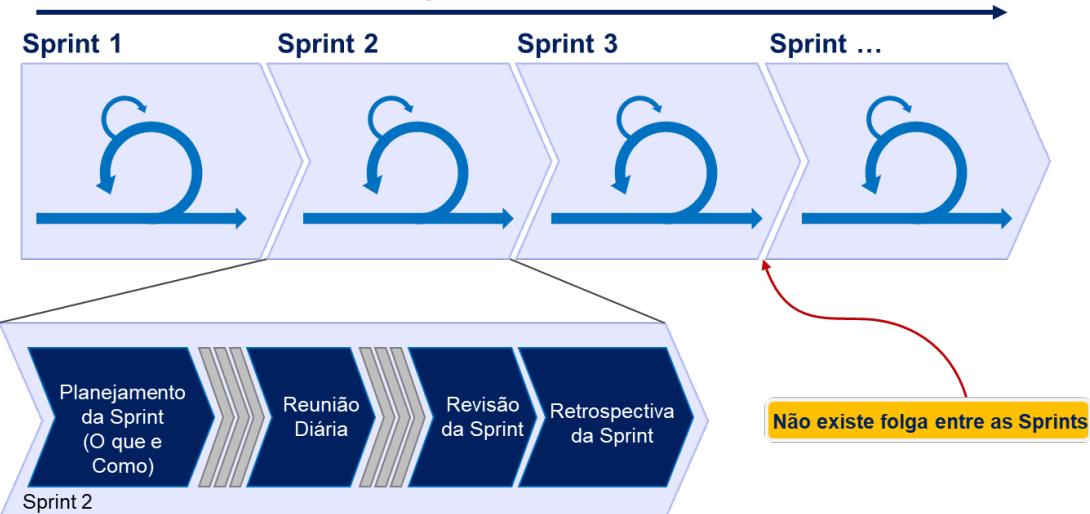
8.2. A Sprint

Tradução de Sprint = corrida de velocidade, arrancada. É o principal evento do Scrum.



A Sprint é um evento que contém todos os outros eventos. Ela se inicia na reunião de Planejamento da Sprint e termina no encontro de Retrospectiva da Sprint.

Desenvolvimento avança ao longo das Sprints



8.3. Definindo a duração (horizonte) da Sprint

A duração da Sprint deve ser de 1 mês ou menos. Alguns times utilizam durações de 1 e outros de 4 semanas. Não há uma regra única e derradeira sobre como definir o tamanho da Sprint. O que é certo para uma pode não ser para outra.

A definição da duração da Sprint deve ser guiada pelos seguintes fatores:

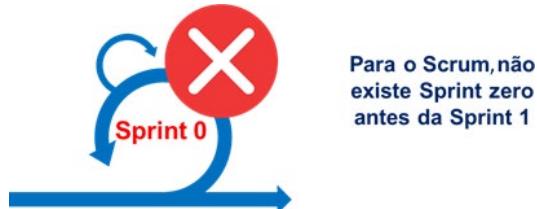
1. O espaço de tempo entre uma liberação e outra.
2. A quantidade de incerteza.
3. A facilidade de obter feedback.
4. Quanto tempo prioridades podem permanecer inalteradas.
5. A sobrecarga (overhead) de cada Sprint.
6. Qual o senso de urgência que se deseja estabelecer.

8.4. Antipadrões de uso das Sprints

Cada Sprint deve produzir ao menos um incremento pronto e potencialmente liberável. Alguns times criam a Sprint zero para preparar o Backlog do Produto e outras coisas necessárias, incluindo configurar o ambiente de trabalho, ferramentas de configuração etc.

Nesse sentido, a Sprint zero não produz nenhum incremento pronto e potencialmente liberável. Se não há incremento pronto, não há oportunidade para feedback, inspeção e adaptação. E sem o feedback não podemos avaliar o retorno sobre o investimento feito durante a Sprint.

Portanto, não existe Sprint 0 para o Scrum ou qualquer outro tipo de Sprint que não produza um incremento verdadeiramente pronto e que possa ser inspecionado.



8.5. Os benefícios do Timebox

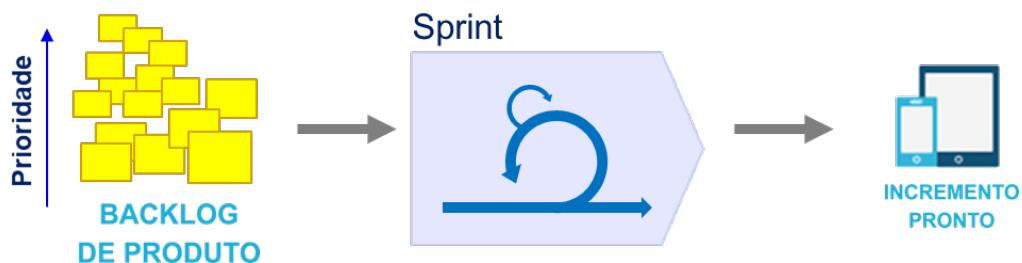
O uso das timeboxes traz alguns benefícios importantes no contexto do Scrum como por exemplo:

- O timebox promove limite para o autogerenciamento. Um time precisa ser autogerenciado, mas isso não é sinônimo de bagunça ou de as pessoas fazerem tudo o que querem. Quando dissemos para um time que ele tem 15 minutos diários para determinada reunião, estamos dando a ele um limite. É possível se organizar do jeito que quiser, desde que a reunião aconteça em 15 minutos. Outro exemplo: a SPRINT que tem no máximo 30 dias. Este é outro limite dado: após 30 dias tem que sair incremento pronto. O time pode fazer do jeito que quiser, desde que entregue o incremento pronto.
- As timeboxes oportunizam regularidade e disciplina, e, em particular, nos ajudam a criar soluções potencialmente liberáveis. Também minimizam a necessidade de reuniões adicionais e evitam as interrupções desnecessárias.
- Promovem o foco (fazer o que é mais valioso no tempo disponível), ajudando a nos concentrarmos no valor e nas necessidades do negócio. A duração máxima é sempre fixa: não podemos estender o tempo para terminar alguma coisa e, em vez disso, temos de compreender quais são as atividades mais valiosas e nos concentrarmos nelas.

9. Planejamento da Sprint

9.1. Visão geral sobre o Planejamento da Sprint

Todo o trabalho realizado pelo Time Scrum para transformar os itens do backlog em incremento de produto pronto e utilizável é feito durante a Sprint e planejado durante a reunião de Planejamento da Sprint. Trata-se então de um encontro do qual todo o Time Scrum participa de maneira colaborativa, para definir o **porquê, o que e o como**.



Quem participa?

Todo o Scrum Team + stakeholders convidados.

Timebox

O Planejamento da Sprint tem um timebox definido com duração máxima de oito horas para uma Sprint de um mês. Para Sprints mais curtas, o evento geralmente é dura menos.

Objetivo

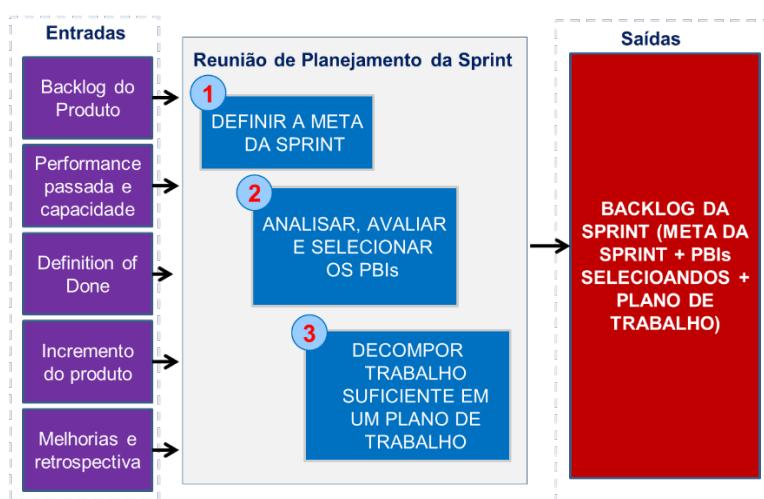
O Dono do Produto garante que os participantes estejam preparados para discutir os itens mais importantes do Backlog do Produto e como eles são mapeados para a Meta do Produto. Um item preparado é aquele que é entendido pelo Time Scrum e cabe em uma única Sprint.

O planejamento da Sprint tem por propósito principal responder a três questões:

- 1 - Por que esta Sprint é valiosa? (PORQUE)
- 2 - O que pode ser feito nesta Sprint? (O QUE)
- 3 - Como o trabalho escolhido será realizado? (COMO)

9.2. O fluxo do Planejamento da Sprint

A seguinte ilustração representa a estrutura de uma reunião de Planejamento da Sprint:



Veja que os itens em cor roxa são os de entrada para este evento, ou seja, é aquilo que todos precisam saber para poderem iniciar a reunião. É importante que todos conheçam a capacidade de trabalho do time e a performance de sprints anteriores, afinal, é neste evento que será decidido quantos e quais itens do backlog do produto serão produzidos. Então, se o time tem capacidade de produzir por exemplo 20 pontos por Sprint, não adianta ter a expectativa de que, para a Sprint atual, ele conseguirá entregar muito mais que isso. O time tem uma capacidade produtiva limitada, ainda que possa ter oscilações de Sprint para Sprint, por exemplo: a adição de dois novos membros.

Outro item que precisa estar preparado é o Backlog do Produto. E o Dono do Produto tem a responsabilidade de levar este backlog ordenado e devidamente detalhado para esta reunião.

Também é necessário ter clareza da Definição de Pronto porque isso guiará os Desenvolvedores quando forem fazer o plano de trabalho.

O incremento de produto pronto também é uma entrada, afinal a sprint atual será feita e agregará mais itens prontos aos já existentes.

E, por fim, é interessante também selecionar uma ou mais melhorias que foram discutidas na retrospectiva passada para que possam ser colocadas em prática na sprint atual. Isso não é obrigatório, mas é altamente desejável.

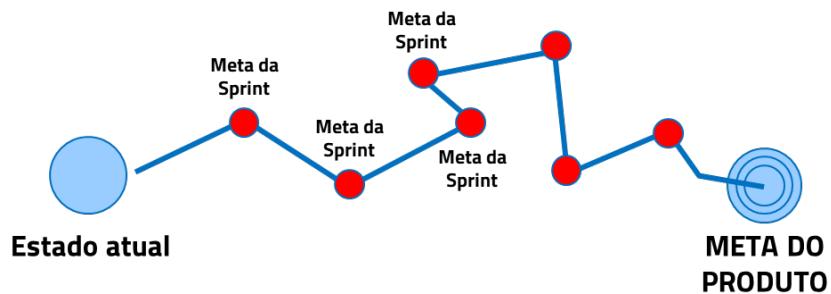
Com base nestas entradas, são discutidos os três tópicos. O planejamento da Sprint tem por propósito principal responder a três questões:

- 1 - Por que esta Sprint é valiosa? (PORQUE)
- 2 - O que pode ser feito nesta Sprint? (O QUE)
- 3 - Como o trabalho escolhido será realizado? (COMO)

Como resultado deste evento, temos o que é chamado de Backlog da Sprint, que é formado pela Meta da Sprint, os itens que foram selecionados do Backlog do Produto e o plano de trabalho.

9.3. Tópico 1 - POR QUE vamos trabalhar nesta sprint?

Tópico um: Por que esta Sprint é valiosa? O Dono do Produto propõe como o produto pode aumentar seu valor e utilidade na Sprint atual. Todo o Time Scrum então colabora para definir uma Meta da Sprint que comunica porque a Sprint é valiosa para os stakeholders. **A meta da Sprint deve ser concluída antes do final da Sprint Planning.**



9.4. Definindo a Meta da Sprint

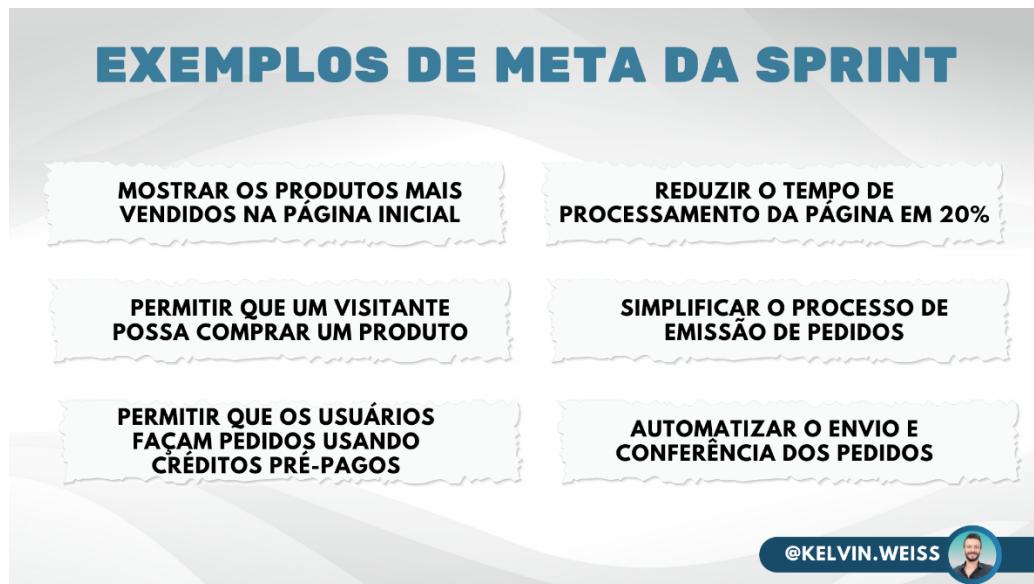
A Meta da Sprint é o único objetivo da Sprint. Embora a Meta da Sprint seja um compromisso dos Desenvolvedores, ela fornece flexibilidade em termos do trabalho exato necessário para alcançá-la. Isso significa que alcançar a meta da sprint é mais importante do que entregar todos os PBIs que foram puxados para a Sprint.

A Meta da Sprint também cria coerência e foco, encorajando o Scrum Team a trabalhar junto ao invés de em iniciativas separadas.

Ela também ajuda os Desenvolvedores a focar e tomar boas decisões, mesmo quando o Dono do Produto não estiver disponível. Sua criação ocorre durante o evento Sprint Planning e então é adicionada ao Sprint Backlog.

A meta definida deve ser uma frase curta que possa ser medida ao final da Sprint. Metas muito genéricas ou filosóficas são difíceis de serem medidas e devem ser evitadas.

EXEMPLOS DE META DA SPRINT



The infographic consists of six rectangular boxes arranged in a 3x2 grid. Each box contains a different goal statement:

- MOSTRAR OS PRODUTOS MAIS VENDIDOS NA PÁGINA INICIAL
- REDUZIR O TEMPO DE PROCESSAMENTO DA PÁGINA EM 20%
- PERMITIR QUE UM VISITANTE POSSA COMPRAR UM PRODUTO
- SIMPLIFICAR O PROCESSO DE EMISSÃO DE PEDIDOS
- PERMITIR QUE OS USUÁRIOS FAÇAM PEDIDOS USANDO CRÉDITOS PRÉ-PAGOS
- AUTOMATIZAR O ENVIO E CONFERÊNCIA DOS PEDIDOS

At the bottom right of the grid is a circular profile picture of a man and the text '@KELVIN.WEISS'.

9.5. Tópico 2 – Em O QUE vamos trabalhar nesta sprint?

Tópico dois: O que pode ser feito nesta Sprint? Por meio de discussão com o Dono do Produto, os Desenvolvedores selecionam itens do Backlog do Produto para incluir na Sprint atual.

O Time Scrum pode refinar esses itens durante este processo, o que aumenta a compreensão e a confiança.

Selecionar o quanto pode ser concluído em uma Sprint pode ser um desafio. No entanto, quanto mais os Desenvolvedores sabem sobre seu desempenho anterior, sua capacidade futura e sua Definition of Done, mais confiantes eles estarão em suas previsões quanto a Sprint.

9.6. Como definir e gerenciar a capacidade de entrega

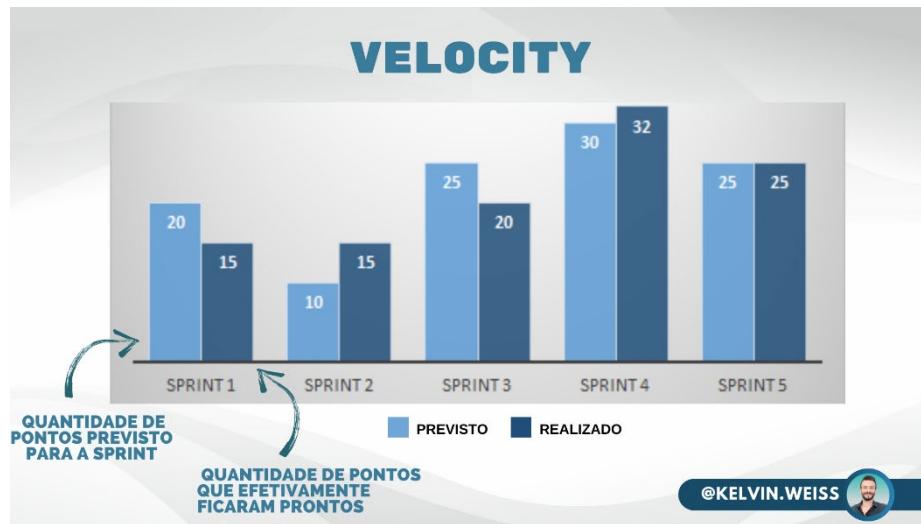
Uma das formas de medir e gerenciar a capacidade de entrega de um Time Scrum é por meio de uma métrica chamada Velocity.

Essa é uma das métricas mais básicas do Scrum. Ela é utilizada para medir a capacidade de entrega do time, sendo obtida a partir da entrega dos Sprints. Velocity mede a quantidade média de trabalho que um time pode concluir em uma Sprint. Ela pode ser medida na forma de pontos da história, itens de trabalho (número de cards) ou até mesmo horas sendo muito útil para a previsão.

Não importa a unidade de medida que você utilizar: número de cards, Story Points, T-Shirt Size (número de itens P, M ou G que o time entrega por Sprint) ou horas. Você tem de conhecer o passo do seu time para que seja possível fazer ajustes quando há variação para menos ou quando se quer acelerar.

Como medir a velocidade do time? Os métodos ágeis são empíricos, logo, baseiam-se em experiência prática passada para poder tirar conclusões sobre o futuro próximo. Sendo assim, o único jeito de saber a

velocidade do time é fazendo com que ele rode de uma a três sprints, de preferência com a mesma formação, para ter uma média e poder obter subsídios para calcular essa métrica.



9.7. Tópico 3 - Definindo o plano de trabalho (COMO) para a Sprint

Tópico três: Como o trabalho escolhido será realizado? Para cada item selecionado do Backlog do Produto, os Desenvolvedores planejam o trabalho necessário para criar um Incremento que atenda à Definition of Done, o que geralmente é feito decompondo itens do Backlog do Produto em itens de trabalho menores que um dia ou menos. A forma como isso é feito fica a critério exclusivo dos Desenvolvedores. Ninguém mais diz a eles como transformar itens do Backlog do Produto em incrementos de valor.



Não há necessidade de os Desenvolvedores identificarem todas as tarefas que precisam ser feitas, apenas o suficiente para prever o trabalho que eles acreditam que poderá ser feito durante a Sprint atual. Para os primeiros dias de execução da Sprint, as tarefas devem ser quebradas idealmente com duração de 1 dia ou menos. As demais podem ter duração maior e os Desenvolvedores precisarão se organizar para detalhar melhor essas tarefas ao longo da Sprint.

9.8. O Backlog da Sprint

O Backlog da Sprint é composto pela Meta da Sprint (por que), o conjunto de itens do Backlog do Produto selecionados para a Sprint (o que), bem como um plano de ação para entregar o Incremento (como).

O Backlog da Sprint é um plano feito por e para os Desenvolvedores. É uma imagem altamente **visível**, em **tempo real** do trabalho que os Desenvolvedores planejam realizar durante a Sprint para atingir a Meta da Sprint.

Consequentemente, o Backlog da Sprint é atualizado ao longo da Sprint conforme mais é aprendido, devendo ter detalhes suficientes para que eles possam inspecionar seu progresso na Daily Scrum.

Sprint Backlog			
Forecast	To-Do	In-Progress	Done
Fix My Profile 5		aliquip	ipsum duis sit ipsum
Filter Service Tickets 8	dolor ipsum culpa	vale culpa	aliquip
Quick Tips 3	ipsum sit duis		

10. Durante a Sprint

10.1. O que é feito durante a Sprint?



Desenvolvimento de Produto

Dentro da Sprint, o Time Scrum passa a maior parte do tempo desenvolvendo produto.

- Transformar os itens do Backlog da Sprint em Incremento Pronto.
- Atingir a meta da Sprint.

Faz parte do Backlog da Sprint.



Refinamento do Backlog do Produto

Preparar itens do Backlog do Produto para as Sprints futuras.

Não faz parte do Backlog da Sprint

O Refinamento do Backlog do Produto é o ato de quebrar e incluir definição adicional aos itens do Backlog do Produto para ter itens menores e mais precisos. Esta é uma atividade contínua para adicionar detalhes, como descrição, ordem e tamanho.

10.2. Como o trabalho é realizado durante a Sprint?

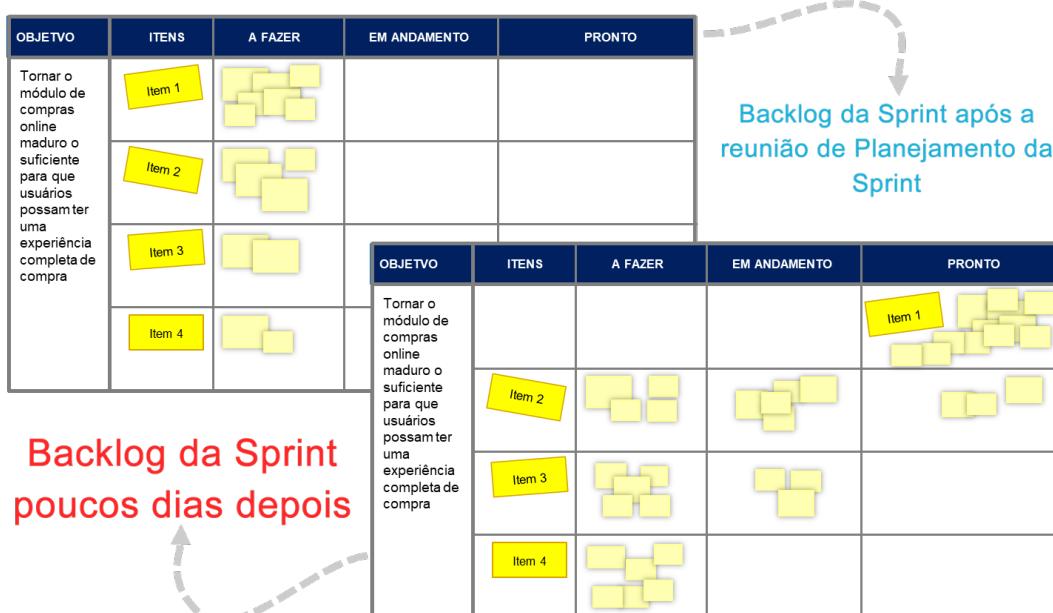
Desenvolvedores são as pessoas do Time Scrum que estão comprometidas em criar qualquer aspecto de um Incremento utilizável a cada Sprint. Desta forma, todos os Desenvolvedores são responsáveis pelo Backlog da Sprint como um todo. Um membro pode “estar” responsável por um determinado item, mas todos “são” responsáveis por tudo.

Os Desenvolvedores são os donos do Backlog da Sprint. Somente eles podem adicionar ou remover tarefas deste Backlog, o que pode ser feito por qualquer membro do Time de Desenvolvimento sempre que for necessário, a qualquer momento.

O Backlog da Sprint torna visível todo o trabalho que os Desenvolvedores identificaram como necessário para atingir a Meta da Sprint. É muito comum que os times criem quadros Scrum para deixar visível o backlog da sprint.

Cada um dos Desenvolvedores escolhe qual ou quais atividades deseja realizar. À medida que um Desenvolver termina sua atividade, ele pega outra disponível no Backlog da Sprint.

É normal que durante a Sprint o Time de Desenvolvimento, conforme vai aprendendo mais sobre o trabalho, refine as tarefas que ainda estão em mais alto nível.



O Time de Desenvolvimento não pode perder de vista o objetivo da Sprint durante a sua realização.

Durante a Sprint:

- Nenhuma mudança que coloque em risco a meta da Sprint é feita;
- A qualidade não diminui;
- O Product Backlog é refinado conforme necessário;
- O escopo pode ser esclarecido e renegociado com o Product Owner conforme mais é aprendido.

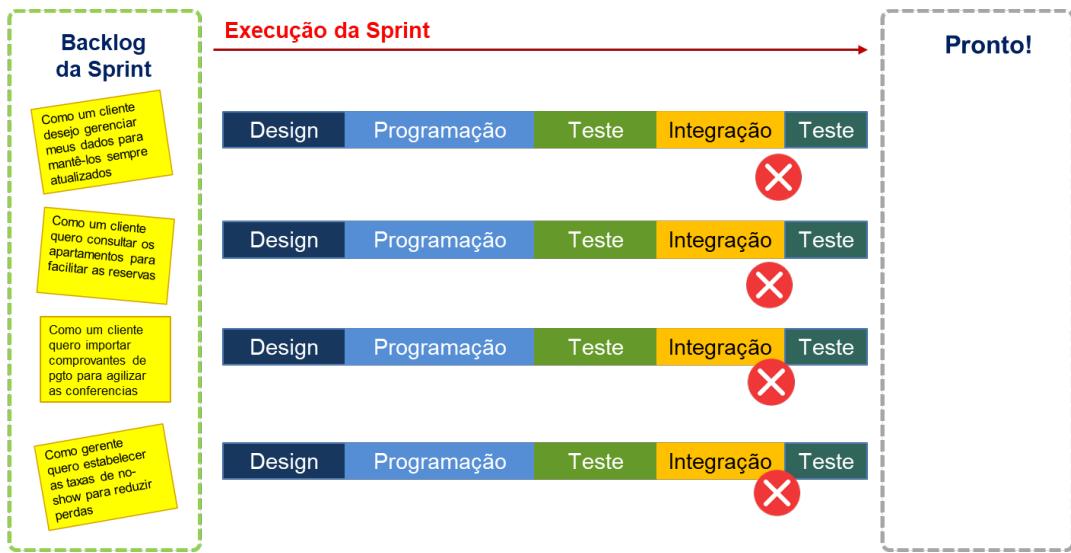
O Dono do Produto deve trabalhar junto com os Desenvolvedores ao longo da Sprint. Quando eles terminam o desenvolvimento de um item, já podem chamar o Dono do Produto para que veja e dê feedback. Um erro comum é deixar o Dono do Produto ver os itens prontos somente na reunião de Revisão da Sprint.

10.3. Como lidar com as mudanças durante a Sprint?

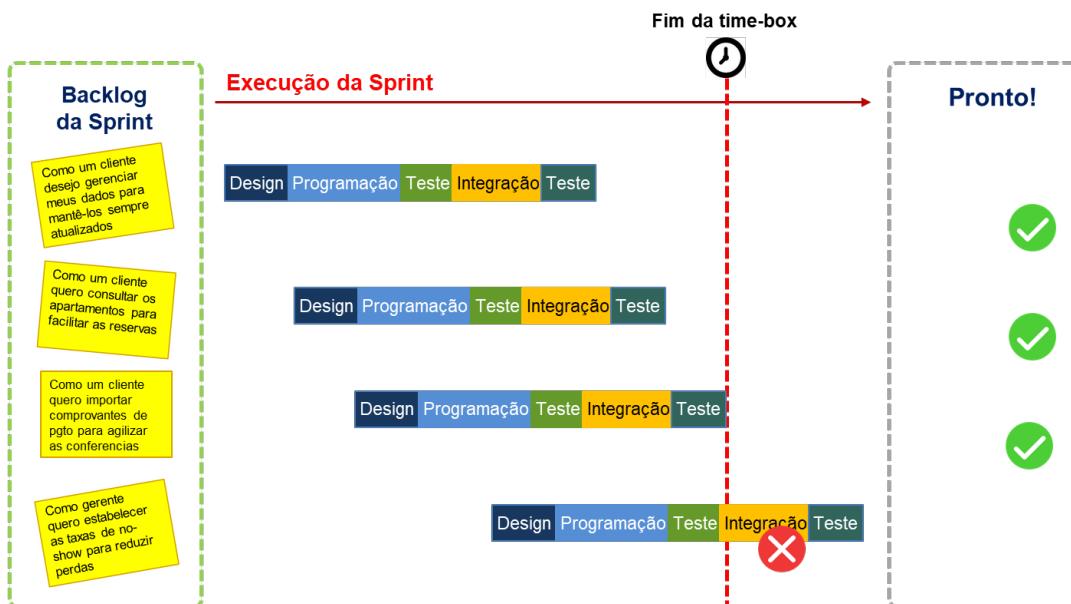
Durante a Sprint, nenhuma mudança que coloque em risco a meta da Sprint é feita. O que é fixo são a qualidade e a meta da sprint. O restante pode ser modificado desde que seja um ato de colaboração entre os membros do Time Scrum.

10.4. Fuja do Water Scrum Fall

Padrão Water Scrum Fall



Padrão Iterativo e Incremental



**O objetivo maior de toda Sprint é gerar um incremento de produto pronto.
A maior falha é não conseguir atingir isto.**

10.5. Gerenciando o trabalho durante a Sprint

É importante que todo o Time Scrum possa a qualquer momento ter uma visão de como está a Sprint, o que está em andamento, o que ainda não foi iniciado e o mais importante: o que ainda falta ser feito (trabalho remanescente).

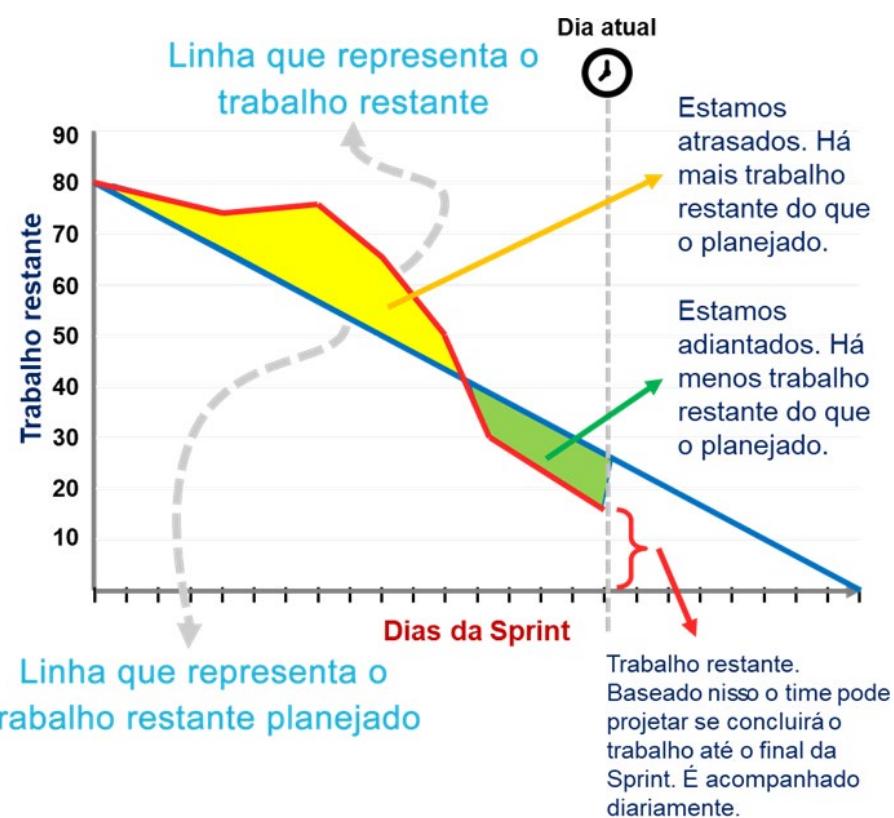
Os Desenvolvedores acompanham o trabalho restante pelo menos uma vez por dia (durante a reunião diária) para projetar a probabilidade de alcançar a Meta da Sprint. Ao acompanhar o trabalho restante ao longo da Sprint, eles podem gerenciar o seu progresso.

Existem várias práticas para prever o progresso, como burn-downs, burn-ups ou cumulative flows. Embora comprovadamente úteis, elas não substituem a importância do empirismo. Em ambientes complexos, o que acontecerá é desconhecido. Somente o que já ocorreu pode ser usado para a tomada de decisão voltada para o futuro.

10.6. Entendendo o gráfico Burn-down

O gráfico Burn-down mostra a quantidade de **trabalho restante** em uma Sprint ao longo do tempo em vez do trabalho concluído. Tem como objetivo mostrar a quantidade de trabalho restante em comparação ao que foi planejado.

Considere que, no Scrum, a duração ou o tempo que se gastou em uma atividade não é fator relevante no contexto. O trabalho restante e a data são as únicas variáveis de interesse.



10.7. A reunião diária

O propósito da Reunião Diária é inspecionar o progresso em direção à Meta da Sprint e adaptar o Backlog dela conforme necessário, ajustando o próximo trabalho planejado.

Reunião diária do Scrum ou Daily Scrum é um evento timeboxed de 15 minutos que deve ocorrer todos os dias úteis durante uma Sprint. O timebox é sempre de 15 minutos, não importando se a duração da Sprint é de 1, 3 ou 4 semanas.

A reunião acontece no mesmo local e horário para criar cadência e reduzir a complexidade. Se o Dono do Produto ou o Scrum Master estão trabalhando ativamente nos itens do Sprint Backlog, eles participam como Desenvolvedores.

O Scrum não especifica horário para a reunião, mas na prática os times tendem a realizá-la na primeira parte da manhã. Se todos começam a trabalhar às oito horas da manhã, um bom horário para começar a reunião é lá pelas 9h, 9h15min. Esse delay entre o início da jornada de trabalho e o começo da reunião diária serve para dar tempo a todos de chegarem, evitando que a reunião seja cancelada ou adiada por atrasos e também é útil porque, antes de irem para a reunião diária, os Desenvolvedores podem atualizar o status de suas atividades.

A reunião é para falar dos problemas e não para buscar soluções. Qualquer problema apresentado deve ser endereçado após o término da reunião. Por exemplo: reunir-se e replanejar atividades a fim de alcançar a Meta da Sprint.

O encontro minimiza a necessidade de reuniões entre o time no restante do dia e melhora a comunicação de todos. As Reuniões Diárias também identificam os impedimentos, promovem a rápida tomada de decisões e consequentemente eliminam a necessidade de outras reuniões.

A Reunião Diária não é o único momento em que os Desenvolvedores podem ajustar seu plano. Eles costumam se reunir ao longo do dia para discussões mais detalhadas sobre a adaptação ou replanejamento do resto do trabalho da Sprint.

10.8. Como realizar a reunião diária?

Os Desenvolvedores podem selecionar qualquer estrutura e técnicas que quiserem desde que seu Scrum Diário se concentre no progresso em direção à Meta da Sprint e produzam um plano de ação para o próximo dia de trabalho. **Isso cria foco e melhora o autogerenciamento.**

A estrutura da reunião é definida pelos Desenvolvedores e pode ser conduzida de diferentes formas desde que estas foquem no progresso em direção ao objetivo da sprint.

Alguns utilizarão perguntas, outros se basearão em discussões. Aqui segue um exemplo (sugestão apenas, não é obrigatório) do que pode ser utilizado:



O Scrum Master é o responsável por garantir que esta reunião aconteça, mas os Desenvolvedores se reportam a eles mesmos e não ao Scrum Master, que deve ensinar o Time de Desenvolvimento como realizar e manter a reunião em 15 minutos.

Uma recomendação é que antes de irem ao encontro diário, os Desenvolvedores atualizem o status de suas atividades.

A reunião diária é um encontro dos Desenvolvedores para os Desenvolvedores. A obrigatoriedade de participação é somente desses profissionais. O Scrum Master deve garantir que a reunião aconteça, **mas sua presença é opcional**. Na verdade, ele só entra em ação quando for solicitado ou quando sentir necessidade.

10.9. Liberação do incremento de produto

No momento em que um item do Backlog do Produto atende à Definição de Pronto, um incremento nasce.

Vários incrementos podem ser criados em uma Sprint. A soma deles é apresentada na Revisão da Sprint, apoiando assim o empirismo. No entanto, um incremento pode ser entregue aos stakeholders antes do final da Sprint. A Revisão da Sprint nunca deve ser considerada um marco para liberar valor.

10.10. Cancelamento da Sprint

Uma Sprint pode ser cancelada antes de seu término caso a Meta da Sprint não faça mais sentido. Isso pode acontecer por razões de:

- Concorrentes
- Negócios
- Tecnologia.

Somente o Dono do Produto tem autoridade para cancelar uma Sprint. Como as Sprints são de curta duração, os cancelamentos geralmente não fazem muito sentido.

Após um cancelamento, conduza uma nova reunião de Planejamento de Sprint. Não espere a data “correta” da reunião.

11. Finalizando a Sprint

11.1. Uma visão geral sobre a Revisão da Sprint

Ao final de cada Sprint, o Time Scrum realiza uma reunião chamada de Revisão da Sprint. É um evento timeboxed de 4 horas para Sprints de 30 dias e proporcionalmente menor de acordo com a duração da Sprint. Se ela for de 2 semanas, então a reunião de Revisão da Sprint deveria ser de 2 horas embora isso seja apenas uma expectativa.

E quem participa dessa reunião? Desenvolvedores, Scrum Master, o Dono do produto e quem mais o Time Scrum convidar. Por exemplo: a gerência sênior, usuários, etc. O Scrum como um todo é muito receptivo a feedbacks e mudanças, mas este é o melhor evento para pessoas externas ao time darem um retorno a respeito do produto. É uma reunião que aceita e incentiva que outras pessoas participem e contribuam para melhorar o produto. Se, por exemplo, o time quiser ouvir a opinião de alguém da área de vendas, essa é a melhor ocasião para eles serem convidados. A audiência principal deste encontro são os stakeholders. É o momento em que o time Scrum está entregando produto pronto aos stakeholders.

Trata-se de uma reunião mais sobre o produto e menos sobre o processo. O foco é discutir sobre o produto e não ficar falando a respeito do que deu certo e errado a nível de processo. Por exemplo: o dono do produto não gostou da usabilidade de determinada funcionalidade. Ele vai explicar o porquê de não ter gostado, mas não é o momento de discutir a causa raiz disso, apontando o que levou o time a fazer uma usabilidade ruim. O Time precisa entender o que não está bom e, mais tarde, durante a reunião de retrospectiva, todos discutem o processo, a razão de a usabilidade ter saído ruim e talvez até mudem a forma de fazer as coisas. Mas na revisão da sprint o objetivo e o foco são o produto.



Time-box de 4 horas
(para sprints de 4 semanas)



Participa todo
o time Scrum



Stakeholders
convidados



Mais sobre o produto,
menos sobre o processo

11.2. Objetivos da Revisão da Sprint

A Revisão da Sprint possui alguns objetivos. Veja a seguir:



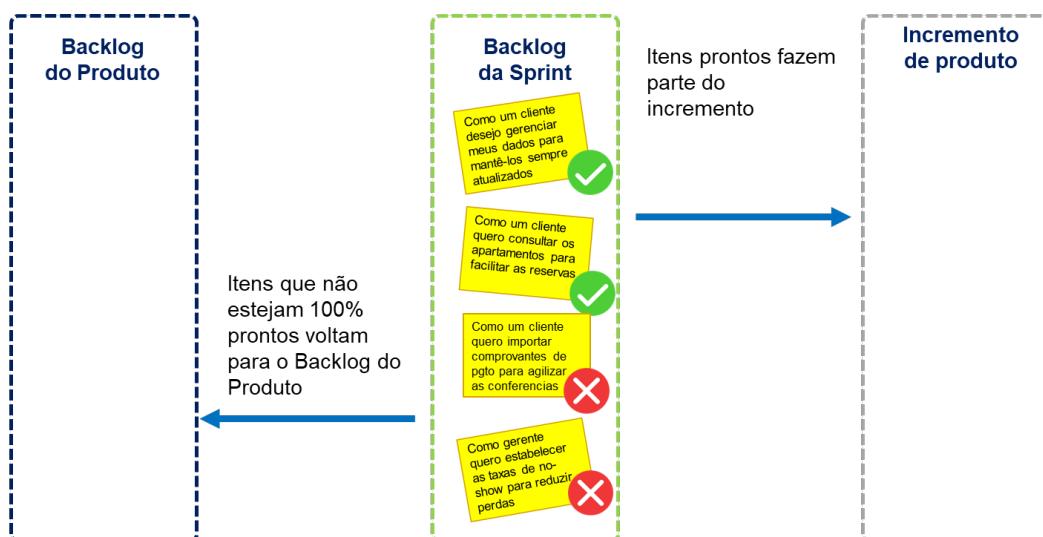
O primeiro deles é demonstrar os itens prontos e obter feedback do Dono de Produto e demais stakeholders. Talvez a maior parte da reunião seja gasta cumprindo este objetivo. É o momento no qual o Time Scrum apresenta os itens do backlog que ficaram prontos e pede feedback para o Dono de Produto e outros convidados. Ou seja, estão exercitando a inspeção do produto.

Demonstrar o produto funcionando é um grande objetivo desta reunião, mas não o único. Com base nos feedbacks que foram dados durante a demonstração, o Dono de Produto pode incluir, alterar ou excluir itens que estejam no backlog de Produto. Ou seja, ele está adaptando o backlog para refletir o feedback dele e dos demais stakeholders convidados para a reunião. Durante o encontro, ele pode ver que, para o produto ficar melhor, serão necessárias funcionalidades adicionais para complementar o que já está pronto, então adiciona isso ao backlog do produto.

O terceiro objetivo é monitorar o progresso do desenvolvimento. Também durante esta reunião, o Dono de Produto apresenta para o time e para os demais stakeholders como está o andamento geral dos trabalhos: quanto do backlog do produto já foi feito e quanto ainda falta fazer. Pode-se também fazer projeções de data de entrega, apresentar custos, etc.

O ideal é que o time tenha concluído todos os itens do Backlog do Produto alocados para a Sprint, mas é mais importante que ele alcance a Meta da Sprint. Talvez um item ou outro pode até não estar completo e isso às vezes acontece realmente, desde que, mesmo assim, a meta da Sprint tenha sido cumprida.

Somente itens prontos devem ser apresentados. Itens que não estejam 100% prontos voltam para o Backlog do Produto.



11.3. Retrospectiva da Sprint

Fundamentalmente uma Sprint deve resultar em duas coisas: um incremento de produto pronto, de qualidade e que atinge as expectativas dos usuários e um time melhor, mais motivado, mais produtivo e mais colaborativo. A Retrospectiva da Sprint é fundamental para que este segundo objetivo seja atingido.



A reunião de Retrospectiva da Sprint tem papel fundamental nesses objetivos, pois permite a **inspeção** do que não está bom, do que pode melhorar e qual o **plano (adaptação)** para melhorar.

A reunião de retrospectiva é um evento timeboxed de 3 horas para uma Sprint de 30 dias. Participa dela todo o Time Scrum. Essa reunião é mais focada no processo e menos no produto.

Basicamente ela tem por objetivo responder a três perguntas:

- O que fizemos de bom e devemos manter?
- O que pode ser melhorado?
- Como vamos implementar as ações para melhorar?

Em outras palavras é: o que funcionou e o que não funcionou? O time deve conversar abertamente sobre os acertos e erros durante a Sprint realizada.

Os objetivos da reunião de Retrospectiva da Sprint são:

- Inspecionar como a última Sprint foi em relação às pessoas, aos relacionamentos, aos processos e às ferramentas.
- Identificar e ordenar os principais itens que foram bem e as potenciais melhorias.
- Criar um plano para implementar melhorias no modo como o Time Scrum faz seu trabalho.

O Scrum Master encoraja o Time Scrum a melhorar dentro dos processos do framework do Scrum e de desenvolvimento, bem como indicando as práticas para fazê-los mais efetivos e agradáveis para a próxima Sprint.

Durante cada Retrospectiva da Sprint, o Time Scrum planeja formas de aumentar a qualidade do produto, adaptando a definição de “pronto” quando apropriado.

Ao final da Retrospectiva da Sprint, o Time Scrum deverá ter identificado melhorias que poderão ser potencialmente implementadas na próxima Sprint. A implementação delas é a forma de adaptação à inspeção que o Time Scrum faz de si próprio (e embora não seja obrigatório, é altamente recomendado).

A Retrospectiva da Sprint fornece um evento dedicado e focado na inspeção e na adaptação, no entanto as melhorias podem ser adotadas a qualquer momento.

O Scrum Master tem papel fundamental nesta reunião: é ele quem deve encorajar o time a falar sobre as coisas boas e ruins que aconteceram durante a Sprint. É responsabilidade dele e de todos os outros membros da equipe construir um ambiente no qual confiança, honestidade, integridade e transparência são fundamentais.

Cada membro do Time Scrum deve dizer abertamente o que considera que foi bem e o que deve ser melhorado. Quando todos tiverem falado e discutido sobre os itens de melhoria, o próprio time prioriza os mais importantes. E para estes, o time conversa sobre possíveis ações para maximizar o resultado do item, caso seja positivo, ou para minimizar o impacto, se for um item negativo.

O Scrum Master está na reunião para ajudar o time a encontrar a melhor maneira de trabalhar unido, sendo produtivo dentro dos princípios do Scrum. Quando esta reunião não resulta em itens de melhoria, torna-se frustrante e perde o seu valor.

12. Planejamento com Scrum

12.1. Uma visão geral sobre Planos e Planejamento

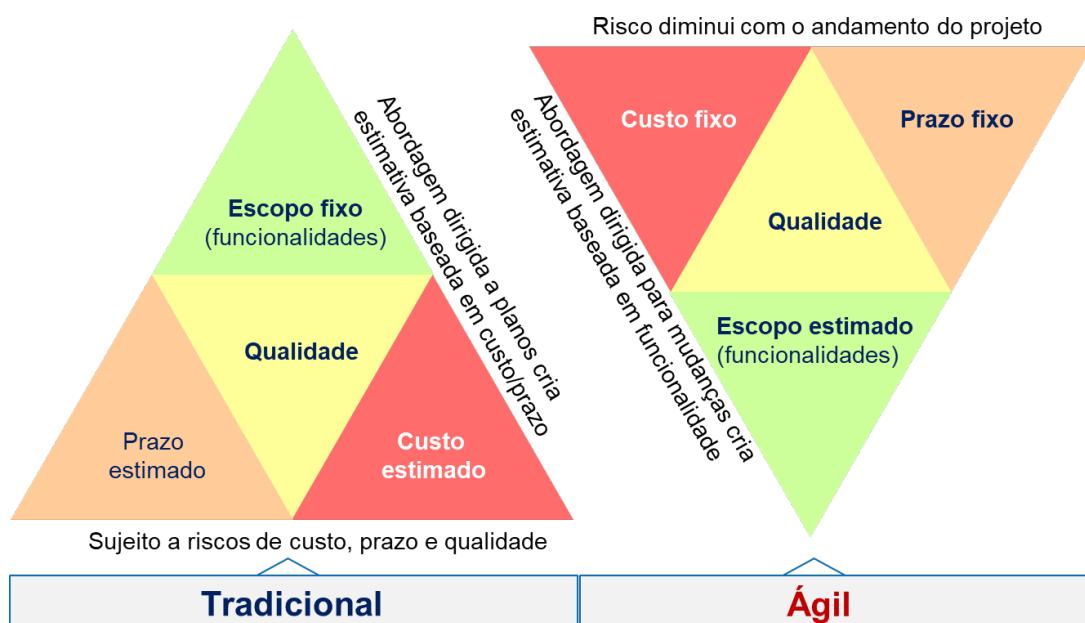
Planos são documentos. Eles são uma “foto” de como acreditamos que o projeto ocorrerá. **Planejamento** é uma atividade. É o processo de analisar informações atuais e projetadas para criar planos de futuro.

Um bom planejamento é aquele no qual partes interessadas encontram informações suficientemente confiáveis e que podem ser usadas como base para tomada de decisões. Posteriormente, para permanecer útil, este plano terá de ser mais preciso.

O planejamento ágil balanceia o esforço e o investimento no planejamento com o conhecimento de que precisaremos revisar o plano ao longo do tempo. Ele muda a ênfase do plano para o planejamento.

12.2. Mudança de mentalidade

Existem três variáveis em qualquer planejamento, considerando que a qualidade não é questionável, ou seja, não podemos alterar o nível de qualidade para entregarmos mais cedo ou mais barato. Estas três variáveis são: **Prazo**, **Custo** e **Escopo**.



No método de planejamento tradicional, é normal que o escopo seja fixo, ou seja, são determinadas logo no início do projeto que funcionalidades devem ser construídas. E, com base nelas, um prazo e um custo estimados são projetados.

O problema disso é que dificilmente conseguimos identificar e compreender tudo o que é necessário no início do projeto. Porém é comum usarmos as definições iniciais para projetar custo e prazo e assumir compromissos com isso. O resultado é quase sempre o mesmo: prazos não são cumpridos e custos quase sempre são maiores. Isso porque ao longo do tempo muitas novidades a respeito do escopo – que a princípio seria fixo – surgem: dificuldades, falta de entendimento, falta de definição do usuário, definições erradas, etc., fazendo com que o prazo e o custo se percam.

A abordagem ágil defende a inversão dessa pirâmide, em que custo e prazo sejam fixos e o escopo passe a ser variável. Ou seja, você define um tempo e um custo e, durante este período e com este dinheiro, faz o melhor que conseguir, entregando o maior valor possível ao cliente em termos de produto. E essa entrega é feita em pedaços pequenos – de forma iterativa e incremental. Assim o cliente pode ir

ajustando o escopo, mudando-o de acordo com suas necessidades para atender mais rápido a elas. Então o custo e o prazo são conhecidos. Você sabe quando o produto será liberado e o custo disso, mas tem apenas uma projeção do escopo que será entregue.

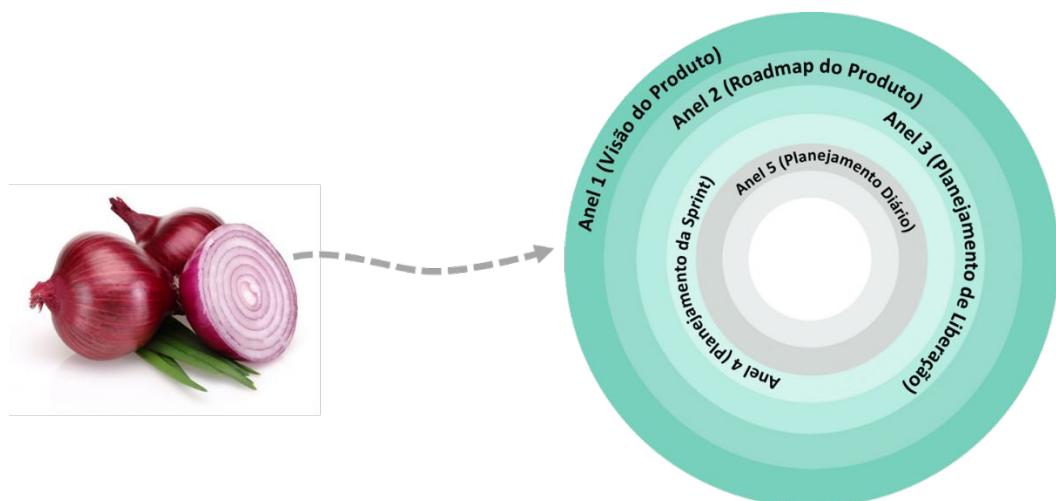
12.3. Múltiplos níveis de planejamento (Planning onion)

Muitos ainda se perguntam como planejar sem investir um tempo muito longo, o que as práticas ágeis melhoram nos planejamentos ou o que muda quando adotamos frameworks como o Scrum, por exemplo.

Entre outras perguntas que sobrevivem ao longo do tempo, estão as clássicas:

- Qual o tempo ideal para se planejar?
- Até que ponto devo detalhar o meu planejamento?
- Qual é o momento efetivo e eficiente para se planejar?

Para ajudar a responder estas dúvidas, precisamos estudar um pouco sobre o planning onion, ou planejamento em multiníveis.

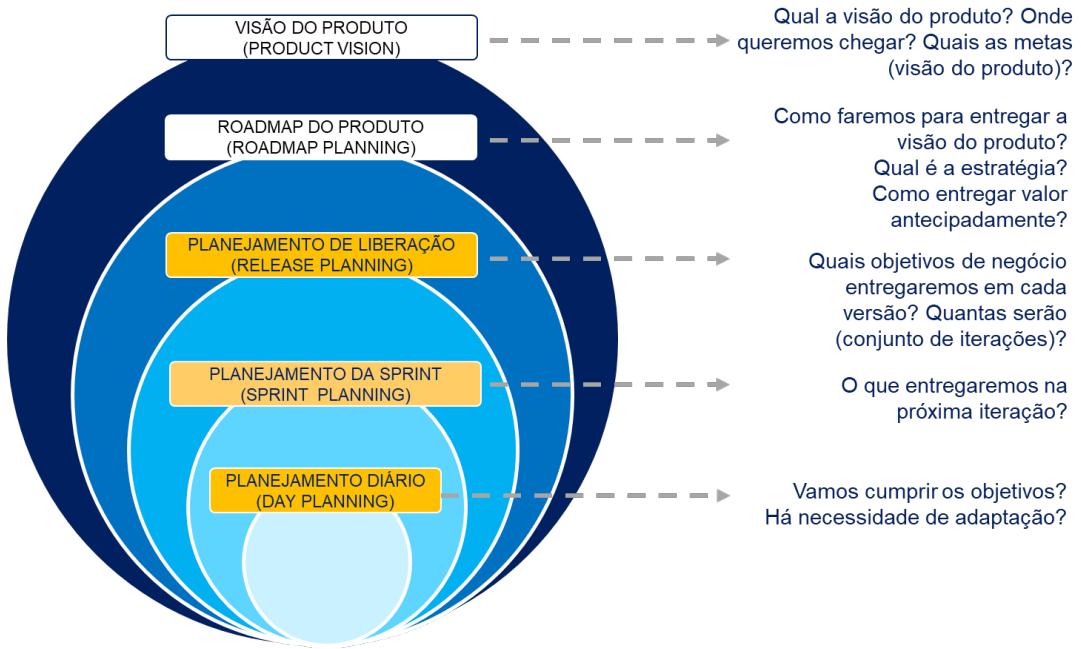


Planning Onion é um termo em inglês que se originou a partir da analogia com os anéis que formam uma cebola, os quais representam os níveis de planejamento que devemos realizar em nossos projetos. Quando lembramos da cebola cortada ao meio e olhamos para os seus anéis, o entendimento fica bem claro se observamos da seguinte maneira:

Os níveis são os momentos em que devemos realizar os planejamentos, os quais têm durações e detalhamentos diferentes. Devemos seguir a ordem de fora para dentro, ou seja, dos anéis maiores para os menores, sendo que analogamente isso significa que o maior é mais superficial, isto é, menos detalhado, e quanto menor for o anel mais detalhado e determinístico deve ser o seu planejamento.

Observe na imagem que temos o nível 1: visão do Produto; depois o nível 2: roadmap do produto; em seguida o nível 3: planejamento de liberação; e nível 4: planejamento da sprint. Por fim, temos o nível 5: planejamento diário.

Outro motivo pelo qual a cebola foi escolhida como analogia para o planejamento, é que ele é cíclico. Isso significa que os níveis de planejamento se retroalimentam. Por exemplo: a partir dos resultados de uma ou de várias sprints, podemos ajustar o planejamento de release, que também pode ser afetado pela definição da visão do produto. Então, embora tenhamos em teoria uma ordem específica para planejar, na prática as coisas se tornam mais cíclicas com feedbacks constantes de um nível de planejamento para o outro.



13. O básico sobre o Scrum em Escala

13.1. O que é Scrum “normal”?

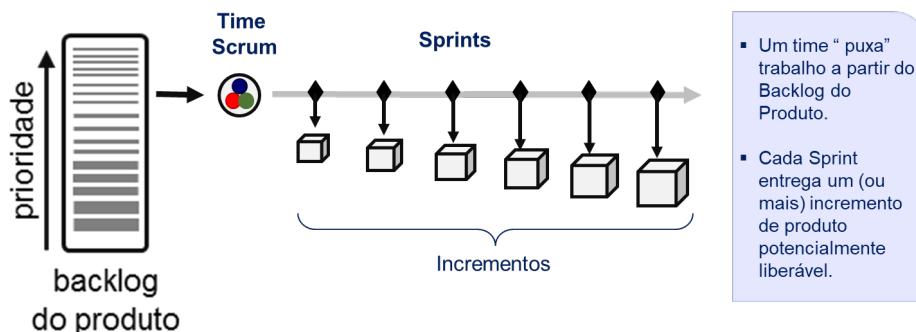
Scrum normal é quando você tem mais de 1 ou, no máximo, 2 times trabalhando no mesmo Backlog do Produto.

Lembrando que a regra mais básica é a seguinte:

Um Produto -> Um backlog de Produto -> Um Dono de Produto -> 1 Time Scrum

Então um Backlog de Produto nunca terá mais de um Dono de Produto. Da mesma forma que um Backlog de Produto sempre representará um único produto.

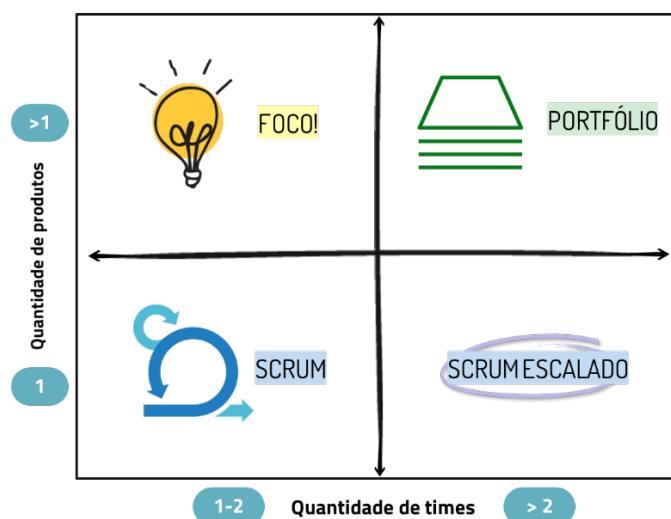
Sendo assim, se existirem dois produtos, existirão dois Backlogs de Produtos e dois Donos de Produtos (um por produto). Lembrando que uma MESMA PESSOA pode eventualmente ser Dono de Produto dos dois produtos e, neste caso, dividirá sua atenção entre eles.



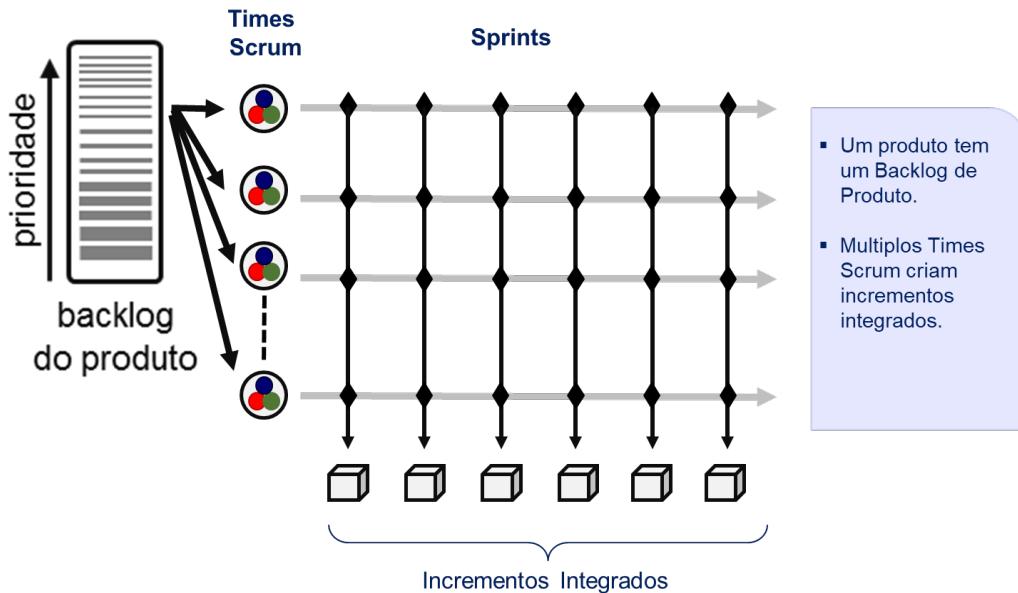
E tudo o que você aprendeu funciona bem para quando há um ou até dois times Scrum trabalhando para um mesmo backlog do produto. Mas e quando é necessário termos mais pessoas trabalhando? 30, 50, 100? É aí que entra o conceito de Scrum escalado.

13.2. O que é Scrum escalado?

Scrum escalado ou Scrum em escala é quando você tem mais de 2 times trabalhando no mesmo Backlog do Produto.



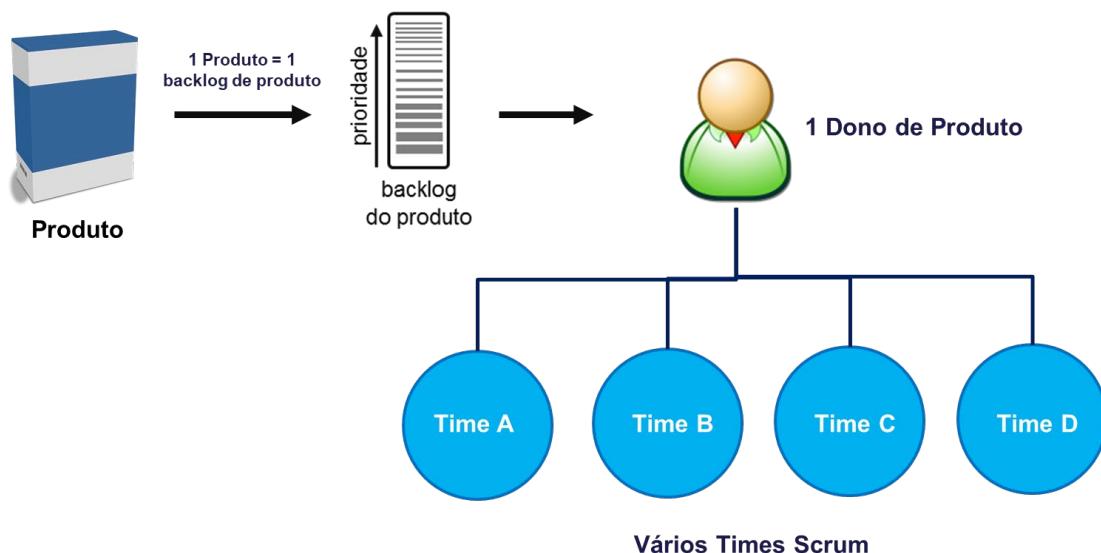
Scrum escalado continua sendo Scrum, portanto todas as regras aprendidas continuam as mesmas. Mas, como ambientes escalados são mais complexos, eles exigem que sejam adicionadas práticas para lidar com essa complexidade adicional. É aqui que entram frameworks de escalação, como o Nexus da Scrum.org, Scrum@Scale, LeSS e SAFe.



Nós não iremos aprender sobre estes frameworks neste treinamento, apenas o básico para dar uma visão geral para você.

13.3. O Dono do Produto em escala

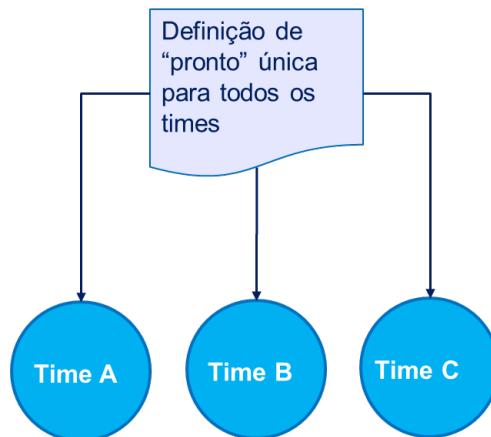
Uma regra básica do Scrum é que somente existirá um Dono de Produto por Backlog de Produto. Isso é imutável. Sendo assim, quando há um produto grande, que exija o trabalho de vários times Scrum, a mesma pessoa será o Dono de Produto de todos esses Times Scrum, ou seja, cada time Scrum tem o seu Dono de Produto, mas esta responsabilidade (papel) é assumida pela mesma pessoa.



13.4. A Definição de Pronto na escala

Se a Definição de Pronto para um incremento faz parte dos padrões da organização, todos os Times Scrum devem segui-la como mínimo. Se não for um padrão organizacional, o Time Scrum deve criar uma Definição de Pronto apropriada para o produto. Os Desenvolvedores devem estar em conformidade com a Definição de Pronto.

Se houver vários Times Scrum trabalhando juntos em um produto, eles devem definir e cumprir mutuamente a **mesma Definição de Pronto**.



Em um ambiente escalado, certifique-se de que:

1. Todos os times que trabalham em associação precisam liberar os incrementos produzidos de forma que se integrem uns aos outros.
2. Assim sendo, todos os times que trabalham integrados devem construir mutuamente a definição de “Pronto”.
3. É absolutamente necessário integrar o trabalho pronto por cada time em suas respectivas Sprints para que o Dono do Produto possa avaliar com precisão o que foi feito.

14. O Scrum Master eficaz

14.1. Scrum Master 1.0

O que se nota no mercado é que muitos profissionais atuam como Scrum Masters 1.0. São pessoas que, embora conheçam bem regras, eventos, papéis e artefatos do Scrum, agem – ainda que de forma inconsciente – com posturas que inibem a auto-organização, o autogerenciamento e o empoderamento do time.

Dentro destas posturas, é possível destacar a de **herói do time**, em que o profissional precisa da adrenalina e do reconhecimento de que é indispensável e a de **Scrum Police**, quando ele força as regras do Scrum ao invés de explicá-las ao time (“a daily passou 5 min, não pode, hein pessoal”, “a discussão está muito boa, mas o timebox acabou. Vamos voltar ao trabalho....”).

Este tipo de postura 1.0 faz com que os times executem um Scrum Mecânico. Ou seja, eles seguem o Scrum, mas você sente que é artificial (reuniões diárias parecem ser todas iguais, melhorias não são colocadas em prática, time sem comprometimento, entre outros efeitos visíveis).

Os times também ficam cada vez mais dependentes do Scrum Master e, quanto mais tempo ele se dedica, mais é necessário, efeito justamente contrário ao que se espera. Quanto mais auto-organizado e empoderado o time for, menos tempo do Scrum Master ele demandará. O foco do profissional 1.0 passa a ser somente o time Scrum e, por consequência, ele não serve ao Dono de Produto e muito menos à Organização.

Dessa forma, impedimentos organizacionais que atrapalham o time não são tratados. A falta desse tratamento e de alguém para tratá-los atrapalha o time no dia a dia, gerando mais trabalho para o Scrum Master e mais dependência. Nota-se então um círculo vicioso e este Scrum Master acaba se tornando ineficaz.

Deve-se então quebrar este círculo e para isso são necessárias novas posturas que elevem o Scrum Master de 1.0 para 2.0, transformando-o em um Scrum Master EFICAZ, aquele que de fato é um líder servidor, que ajuda a empoderar o time e a realizar um Scrum autossustentável.

14.2. Scrum Master Eficaz

Para ser um Scrum Master EFICAZ, é necessário agir com posturas mais adequadas, como **Coach**, **Mentor**, **Professor**, **Removedor de Impedimentos** e **Agente de Mudanças**.

O seu principal papel é tornar os **problemas transparentes** e não necessariamente **resolvê-los**.

Apenas como exemplo, tente o seguinte: não chame a atenção de alguém que não compareceu na reunião diária. Ao invés disso, durante a retrospectiva, pergunte ao time se algo poderia ter ido melhor se todos tivessem participado das reuniões diárias. Ensine o time a resolver seus próprios conflitos e a confiar uns nos outros. Isso fará com que ele fique mais empoderado e autogerenciado, liberando o Scrum Master para atuar mais no nível da organização, ajudando a remover os impedimentos organizacionais que inibem a agilidade.

Conforme definido no Scrum Guide, o Scrum Master deve servir aos Desenvolvedores, ao Dono de Produto e à Organização. Como muitos agem como Scrum Master 1.0, acabam não atuando no nível organizacional, criando dois mitos: o de que Scrum Masters trabalham no nível do time e Agile Coachs no nível organizacional e que o Scrum Master é um Agile Coach júnior.

Scrum Master não é Agile Coach Júnior e Agile Coach não é a evolução do Scrum Master.

Estes profissionais podem até ter atuação diferente, mas trabalham em uma mesma camada. Agile Coach não é a evolução do Scrum Master, *mas isto é assunto para outro momento*.



Quando o Scrum Master atua como Scrum Master EFICAZ, potencialmente alguns efeitos poderão ser notados, como a redução do time-to-market, aumento da felicidade do time, um Scrum autossustentável e o aumento da satisfação de clientes.

Scrum Master é então um fator chave para um ambiente de Scrum Profissional.



15. Encerramento

Esta apostila foi escrita por **Kelvin Weiss (@kelvin.weiss)** com base em seus conhecimentos práticos e diversas fontes de pesquisa e inspiração.

Este material faz parte do curso **O Scrum Master Eficaz**, vendido e distribuído pela **TIEXAMES**.

O material contido aqui não deve ser usado como única fonte de preparação para obter a certificação Professional Scrum Master I da Scrum.org, pois apenas complementa as aulas gravadas do curso O Scrum Master Eficaz.

Obrigado pela ler este material e nos vemos por aí!