

## SCRUM

### Conceitos Básicos

INCIDÊNCIA EM PROVA: ALTA

Galera, alguém de vocês sabe de onde vem esse nome? Então, eu vou contar para vocês! **Esse nome vem do Rugby e é utilizado como uma metáfora para refletir o alto grau de cooperação necessária para obter sucesso no alcance de algum objetivo.** Imagino que poucos de vocês entendam as regras desse esporte, portanto vou explicar de forma bastante rápida e objetiva o porquê dessa metáfora ser utilizada.

No Rugby, um time pontua sempre que a bola cruza a linha de gol e toca o chão – sendo carregada ou por meio de passes. Caso o jogador seja derrubado, ele deve soltar a bola, e a jogada se reiniciará! **Além disso, deve haver intensa troca de passes entre os jogadores, de modo a deixá-los menos vulneráveis a serem derrubados por outros jogadores.** *Calma que tudo isso que estou dizendo fará sentido...*



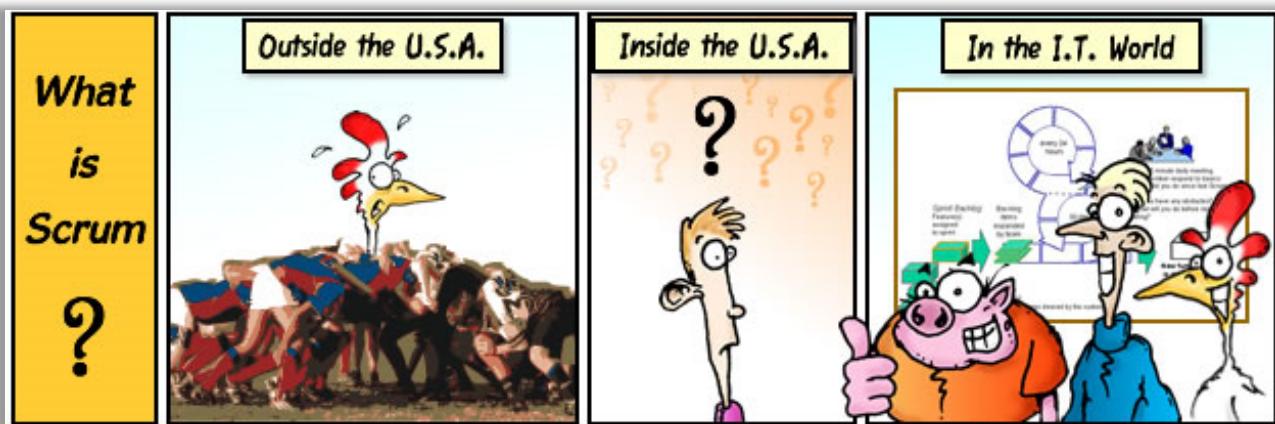
Bem... cada jogada se inicia quando um *scrum* é realizado, isto é, forma-se uma parede de força entre os jogadores, como pode ser visto nas imagens acima. Observem que os jogadores se reúnem de forma bastante próxima e coesa, **unindo suas forças e habilidades para trabalhar em conjunto**



e harmonicamente a fim de conseguir recuperar a bola. Percebam, portanto, que o time inteiro deve trabalhar para que a equipe possa pontuar.

Diferentemente do Futebol Americano, não há um *quarterback* ou uma estrela no time – todos têm suas funções e responsabilidades, e são igualmente importantes! Acredito que agora ficou mais fácil entender de onde vem esse nome. **Antes de iniciarmos a teoria, eu gostaria de enfatizar a importância de ler o guia oficial:** ele tem apenas 13 páginas em sua última versão, mas eu inseri na aula tanto o Guia Scrum Versão 2017 quanto à Versão 2020.

Logo, eu recomendo que vocês o leiam por inteiro, porque **ele é a fonte de praticamente tudo que veremos aqui!** Tem versão em português, é gratuito, é enxuto, então não tem desculpas :)



**Scrum é um framework leve, simples de entender e extremamente difícil de dominar, para desenvolver e manter produtos complexos e adaptativos**, enquanto entrega produtiva e criativamente produtos com o mais alto valor possível. Na minha época de concurso, eu decorava essa definição – sim, eu recomendo decorar algumas definições! Fiquem calmos porque nós vamos esmiuçar cada parte desse conceito.

#### [Guia Scrum - Versão 2017]

Scrum é um framework para desenvolver, entregar e manter produtos complexos. Este guia contém a definição do Scrum. Esta definição consiste em papéis, eventos, artefatos e as regras do Scrum que unem os demais e os mantêm integrados. Ken Schwaber e Jeff Sutherland desenvolveram o Scrum; o Guia do Scrum é escrito e fornecido por eles. Juntos, eles apoiam o Guia do Scrum.

Scrum: um framework dentro do qual pessoas podem tratar e resolver problemas complexos e adaptativos, enquanto produtiva e criativamente entregam produtos com o mais alto valor possível. Scrum é: leve, simples de entender e difícil de dominar.

Scrum é um framework estrutural que está sendo usado para gerenciar o trabalho em produtos complexos desde o início de 1990. Scrum não é um processo, técnica ou um método definitivo. Em vez disso, é um framework dentro do qual você pode empregar vários processos ou técnicas. O Scrum deixa clara a eficácia relativa de suas práticas de gerenciamento de produto e técnicas de trabalho, de modo que você possa continuamente melhorar o produto, o time e o ambiente de trabalho.



O framework Scrum consiste de times Scrum associados a papéis, eventos, artefatos e regras. Cada componente dentro do framework serve a um propósito específico e é essencial para o uso e sucesso do Scrum. As regras do Scrum integram os papéis, eventos e artefatos, administrando as relações e interações entre eles. As regras do Scrum são descritas ao longo deste documento. Estratégias específicas para o uso do framework Scrum variam e são descritas em outros documentos.

#### [Guia Scrum - Versão 2020]

Scrum é um framework leve que ajuda pessoas, times e organizações a gerar valor por meio de soluções adaptativas para problemas complexos.

Em suma, Scrum requer um Scrum Master para promover um ambiente onde:

1. Um Product Owner ordena o trabalho para um problema complexo em um Product Backlog.
2. O Scrum Team transforma uma seleção do trabalho em um incremento de valor durante uma Sprint.
3. O Scrum Team e seus stakeholders inspecionam os resultados e se ajustam para a próxima Sprint.
4. Repita

**Em primeiro lugar, percebam que ele é um framework – isso significa dizer que ele agrupa processos, métodos e técnicas.** Fundamentalmente, ele possui pressupostos, conceitos, valores e práticas, mas quem utilizá-lo pode incluir outras novidades. Ele não te dirá tudo o que fazer, logo você tem a liberdade para fazer o que melhor funcionar dentro das suas necessidades específicas.

Observem que o Scrum é aplicado no gerenciamento do “trabalho” de desenvolvimento dos produtos, e não exatamente para gerenciar o produto em si. Ele é eficaz nas suas práticas de gerenciamento de produto e técnicas de trabalho e se foca na melhoria contínua do produto, do time e do ambiente de trabalho. **Ele não é um método definitivo, ou seja, ele pode ser utilizado em conjunto, ou complementado, por outras práticas, ferramentas e abordagens.**

Em segundo lugar, ele é um documento bastante enxuto conforme acabamos de ver! Vocês verão que, **oficialmente e obrigatoriamente**, ele é composto por três papéis, quatro eventos, três artefatos e por um fluxo (chamado Sprint) – veremos em detalhes adiante. Ele pode comportar outros artefatos, como Gráfico de Burndown, Documento de Visão, etc. No entanto, esses outros não são obrigatórios – são como “plug-ins” que podem ser adicionados.

#### [Guia Scrum - Versão 2017]

O Scrum 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; e,  
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 em nosso dia-dia nas nossas vidas, como indivíduos e sociedades. Como tecnologia, mercado, complexidades ambientais e suas interações têm aumentado rapidamente, a utilidade do Scrum em lidar com a complexidade é provada diariamente.



Scrum demonstra efetividade especialmente na transferência de conhecimento iterativo e incremental. Scrum é agora amplamente usado para produtos, serviços e no gerenciamento da própria empresa. A essência do Scrum é um pequeno time de pessoas. O time individual é altamente flexível e adaptativo. Essas forças continuam operando em únicos, muitos, vários, e em redes de times que desenvolvem, liberam, operam e sustentam o trabalho e trabalham produtos de milhares de pessoas.

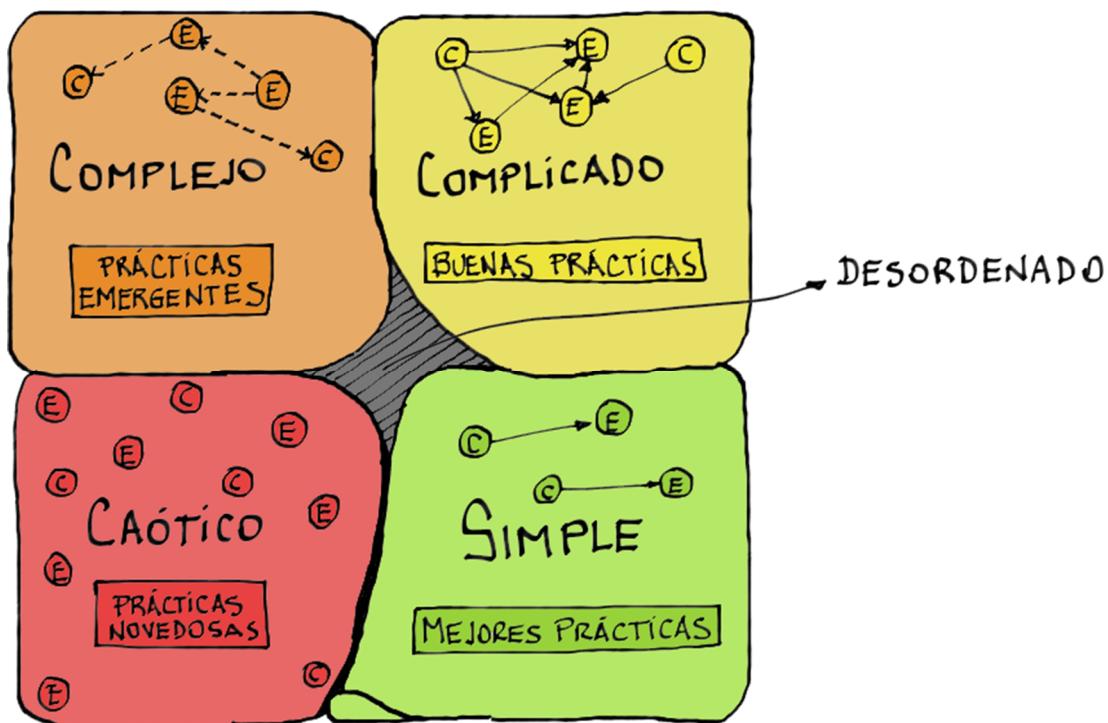
Eles colaboram e interoperam através de arquiteturas sofisticadas de desenvolvimento e ambientes de liberação como objetivo. Quando as palavras "desenvolver" e "desenvolvimento" são usadas no Guia Scrum, elas se referem a trabalho complexo, tais como os tipos identificados acima.

#### [Guia Scrum - Versão 2020]

Scrum é simples. Experimente como está e determine se sua filosofia, teoria e estrutura ajudam a atingir objetivos e criar valor. O framework Scrum é propositalmente incompleto, apenas definindo as partes necessárias para implementar a teoria Scrum. O Scrum é construído sobre a inteligência coletiva das pessoas que o utilizam. Em vez de fornecer às pessoas instruções detalhadas, as regras do Guia do Scrum orientam seus relacionamentos e interações.

Vários processos, técnicas e métodos podem ser empregados com o framework. Scrum se acopla as práticas existentes ou as torna desnecessárias. Scrum torna visível a eficácia relativa da gestão atual, meio ambiente e técnicas de trabalho, para que melhorias possam ser feitas.

O Scrum é um framework para gerenciar projetos, produtos e processos, focado na adaptação em vez de planejamento, que não utiliza muita documentação e que adota processos mais simplificados, facilitando a adaptação às mudanças de requisitos e permitindo entregas rápidas e menores. **Ele é usado em ambientes complexos, onde os requisitos e as prioridades mudam constantemente.**



*Diego, o que seria exatamente um ambiente complexo?* Existe uma coisa chamada **Modelo Cynefin** que explica bem os tipos de ambientes, dividindo-os em simples, complicados, complexos e caóticos. O que vocês precisam saber é que um ambiente complexo é aquele que não é muito bem definido, não é muito acoplado, há muitas mudanças, apresenta muitas formas de realizar um trabalho.

Vamos ver um exemplo: *McDonalds é um ambiente complexo?* Não, é um ambiente simples! Ele é muito bem definido, extremamente acoplado, não tem liberdade e não existem muitas opções de como realizar um trabalho. Em qualquer lugar do mundo, o cardápio será praticamente o mesmo; o cara que faz o sanduba realiza os mesmos passos; não há mudanças; não há várias formas de realizar um trabalho.

O Scrum é um framework em que podem ser empregados vários processos e técnicas. Pode ser definido como um conjunto de papéis, eventos, artefatos e regras associadas a uma equipe. Ele é fundamentado em teorias empíricas de controle de processo e emprega uma abordagem iterativa e incremental (maximizando as oportunidades de feedback) para aperfeiçoar a previsão e controle de riscos.



## Pilares Fundamentais

INCIDÊNCIA EM PROVA: BAIXA

O Scrum se baseia no empirismo e no *lean thinking*. O empirismo afirma que o conhecimento vem da experiência e da tomada de decisões com base naquilo que é verdadeiro e conhecido. Para tal, ele emprega uma abordagem iterativa e incremental para aperfeiçoar e otimizar a previsibilidade e controle de riscos. Já o Lean Thinking é uma espécie de estrutura mental (*mindset*) que permite reduzir o desperdício e se concentrar no essencial.

O Scrum combina quatro eventos formais para inspeção e adaptação, contidos dentro de uma sprint (que eventualmente é considerada por questões de prova como um tipo de evento). Esses eventos funcionam porque implementam três pilares fundamentais para controle do processo empírico: **Transparência**, **Inspeção** e **Adaptação** (é o famoso **TIA**). Vejamos adiante em mais detalhes...



### Transparência

Esse pilar trata de aspectos significativos (e padronizados) devem estar visíveis aos responsáveis pelos resultados. Deve haver transparência dentro e fora da equipe, permitindo a qualquer pessoa compreender o que realmente está ocorrendo, ocasionando melhor comunicação e confiança. Um dos autores, Ken Schwaber, diz: "*Scrum é igual sogra: chega na sua casa e esfrega todos os seus problemas na sua cara*".

Se uma iteração falhar, todos devem ficar sabendo; se os feedbacks forem ruins, todos devem ficar sabendo; se o projeto atrasou, todos devem ficar sabendo. **O objetivo é encarar as dificuldades de forma honesta e chegar a um consenso sobre como estes podem ser ultrapassados**. Os erros são inevitáveis e a equipe deve ser incentivada a encarar esta premissa como uma base para aprender com os erros que são cometidos.

#### [Guia Scrum - Versão 2017]

Aspectos significativos do processo devem estar visíveis aos responsáveis pelos resultados. A transparência requer que estes aspectos tenham uma definição padrão comum para que os observadores compartilhem um mesmo entendimento comum do que está sendo visto. Por exemplo: uma linguagem comum referindo-se ao processo deve ser compartilhada por todos os participantes; e aqueles que realizam o trabalho e aqueles que inspecionam o incremento resultado do trabalho devem compartilhar uma definição comum de "Pronto".

#### [Guia Scrum - Versão 2020]



O processo emergente e o trabalho devem ser visíveis tanto para quem executa o trabalho quanto para quem recebe o trabalho. Com o Scrum, decisões importantes são baseadas no estado percebido de seus três artefatos formais. Artefatos com baixa transparência podem levar a decisões que diminuem o valor e aumentam o risco. A transparência permite a inspeção. A inspeção sem transparência é enganosa e gera desperdício.

## Inspeção

Também chamada de verificação, os usuários devem frequentemente inspecionar os artefatos produzidos e o progresso para detectar variações indesejáveis (claro, não pode ser extremamente frequente ao ponto de atrapalhar a execução das tarefas). Uma vez que todos os problemas sejam transparentes, esse é o momento de inspecionar o processo e o produto em busca de resolver o problema.

**Galera, a ideia aqui é identificar rapidamente qualquer desvio em relação à meta que deve ser atingida.** Nós veremos mais à frente que, tanto na Reunião de Revisão quanto na Reunião de Retrospectiva, os produtos e os processos serão devidamente inspecionados. Essas inspeções em geral são mais benéficas quando realizadas de forma diligente e cuidadosa por inspetores especializados no trabalho a se verificar.

### [Guia Scrum - Versão 2017]

Os usuários Scrum devem, frequentemente, inspecionar os artefatos Scrum e o progresso em direção ao objetivo da Sprint para detectar variações indesejadas. Esta inspeção não deve ser tão frequente que atrapalhe o objetivo dos trabalhos. As inspeções são mais benéficas quando realizadas de forma diligente por inspetores especializados no trabalho a se verificar.

### [Guia Scrum - Versão 2020]

Os artefatos do Scrum e o progresso em direção às metas acordadas devem ser inspecionados com frequência e diligência para detectar variações ou problemas potencialmente indesejáveis. Para ajudar na inspeção, o Scrum fornece cadência na forma de seus cinco eventos. A inspeção habilita a adaptação. A inspeção sem adaptação é considerada inútil. Os eventos Scrum são projetados para provocar mudanças.

## Adaptação

**Se um inspetor determina que um ou mais aspectos de um processo desviou para fora dos limites aceitáveis, e que o produto resultado será inaceitável, o processo ou o artefato sendo produzido deve ser ajustado.** O ajuste deve ser realizado o mais breve possível para minimizar mais desvios. Como mudanças sempre ocorrem, é recomendável se adaptar a mudanças em vez de evitar mudanças.

### [Guia Scrum - Versão 2017]



Se um inspetor determina que um ou mais aspectos de um processo desviou para fora dos limites aceitáveis, e que o resultado do produto será inaceitável, o processo ou o material sendo produzido deve ser ajustado. O ajuste deve ser realizado o mais breve possível para minimizar mais desvios. O Scrum prescreve quatro Eventos formais para inspeção e adaptação, como descrito na seção Eventos do Scrum deste documento: Planejamento da Sprint; Reunião diária; Revisão da Sprint; e Retrospectiva da Sprint.

#### [Guia Scrum - Versão 2020]

Se algum aspecto de um processo se desviar para fora dos limites aceitáveis ou se o produto resultante for inaceitável, o processo que está sendo aplicado ou os materiais que estão sendo produzidos devem ser ajustados. O ajuste deve ser feito o mais rápido possível para minimizar novos desvios. A adaptação se torna mais difícil quando as pessoas envolvidas não são empoderadas ou auto-gerenciadas. Espera-se que um Scrum Team se adapte no momento em que aprende algo novo por meio da inspeção.

**Vamos resumir os três pilares que sustentam o nosso framework! Tudo no Scrum deve ser transparente e facilmente acessível.** Partindo dessa premissa, podemos inspecionar e identificar problemas e oportunidades de melhoria do produto e/ou processo – em geral, por meio de eventos (também chamados de reuniões ou cerimônias). Feito isso, deve-se buscar ajustar e adaptar produto e/ou processo para minimizar desvios.

Pessoal, lembrem-se que o Scrum é um framework para gerenciar projetos! *Em algum momento eu falei aqui sobre desenvolvimento de software? Não, ele é capaz de gerenciar qualquer projeto que vise aumentar a agilidade e qualidade da sua execução.* Embora tenha sido concebido inicialmente como uma metodologia de desenvolvimento de software, ele contém elementos que podem ajudar a formar uma equipe de alto desempenho para qualquer projeto.

**O Scrum apenas fornece um framework estruturado para executar alguns princípios.** As equipes de desenvolvimento de software seguem essa metodologia porque seu trabalho é altamente complexo, interdependente e acelerado. No entanto, se ele funciona para essas equipes, certamente pode funcionar para outros casos. Se você é um líder de uma equipe que gerencia projetos complexos, é interessante pensar na aplicação do Scrum!

Dito isso, notem que todos os conceitos que veremos daqui para frente podem ser aplicados a qualquer projeto, apesar de ter sido concebido para desenvolvimento de software. O primeiro e talvez principal conceito seja a Sprint! *O que é uma Sprint? Trata-se de uma unidade de trabalho que satisfaz um requisito de negócio.* Em outras palavras, é um ciclo completo de desenvolvimento de um incremento potencialmente entregável de um produto.

PILARES	DESCRÍÇÃO
TRANSPARÊNCIA	Todo trabalho deve ser claramente definido e conhecido por todas as partes envolvidas no projeto.
INSPEÇÃO	Todo trabalho deve ser inspecionado com a frequência necessária para garantir a qualidade do produto.
ADAPTAÇÃO	O projeto deve ser capaz de se adaptar ao projeto às necessidades de negócios.



## Principais Valores

INCIDÊNCIA EM PROVA: BAIXA



O Scrum possui basicamente cinco valores principais: **coragem, foco, comprometimento, respeito e abertura**. O sucesso do uso desse framework de gerenciamento de projetos dependerá intrinsecamente de como os membros se tornam mais familiarizados e proficientes em relação a cada um desses valores. Vamos falar logo abaixo sobre cada um deles com um pouco mais de detalhes. Vejamos:

VALORES	DESCRIÇÃO
CORAGEM	Os integrantes de um projeto precisam ter coragem para fazer a coisa certa e trabalharem juntos removendo impedimentos, buscando soluções.
FOCO	Os integrantes de um projeto precisam focar no trabalho durante a sprint e nas metas designadas – time disperso perde produtividade e não alcança os objetivos.
COMPROMETIMENTO	Os integrantes se comprometem com o trabalho que se responsabilizou em fazer, envolvendo-se e não abandonando pela metade ou entregando sem qualidade.
RESPEITO	Os integrantes se respeitam entre si a fim de manter a colaboração, a integração e o bom ambiente de trabalho.
ABERTURA	Os integrantes devem poder ser francos, expor ideias e propostas mesmo que elas não sejam proveitosas. Momentos de debates, discussões e sugestões são ideais.

[Guia Scrum - Versão 2017]

Quando os valores de comprometimento, coragem, foco, abertura e respeito são incorporados e vividos pelo Time Scrum, os pilares do Scrum de transparência, inspeção e adaptação tornam-se vivos e constroem a confiança para todos. Os membros do Time Scrum aprendem e exploram estes valores à medida que trabalham com os eventos, papéis e artefatos



*do Scrum. O Sucesso no uso do Scrum depende das pessoas se tornarem mais proficientes na vivência destes cinco valores.*

*As pessoas se comprometem pessoalmente em alcançar os objetivos do Time Scrum. O Time Scrum precisa ter coragem para fazer a coisa certa e trabalhar em problemas difíceis. Todos focam no trabalho da Sprint e nos objetivos do Time Scrum. O Time Scrum e seus Stakeholders concordam em estarem abertos a todo o trabalho e aos desafios com a execução dos trabalhos. Os membros do Time Scrum respeitam uns aos outros para serem pessoas capazes e independentes.*

#### **[Guia Scrum - Versão 2020]**

*O Scrum Team se compromete a atingir seus objetivos e suportar uns aos outros. Seu foco principal é o trabalho da Sprint para fazer o melhor progresso possível em direção a essas metas. O Scrum Team e seus stakeholders são abertos quanto ao trabalho e os desafios. Os membros do Scrum Team se respeitam quanto a serem pessoas capazes e independentes, e são respeitados como tal pelas pessoas com quem trabalham. Os membros do Scrum Team têm a coragem de fazer a coisa certa e trabalhar em problemas difíceis.*

*Esses valores orientam o Scrum Team em relação ao seu trabalho, ações e comportamento. As decisões que são tomadas, os passos dados e a forma como o Scrum é usado devem reforçar esses valores, não diminuí-los ou miná-los. Os membros do Scrum Team aprendem e exploram os valores à medida que trabalham com os eventos e artefatos do Scrum. Quando esses valores são incorporados pelo Scrum Team e pelas pessoas com quem trabalham, os pilares empíricos do Scrum de transparência, inspeção e adaptação ganham vida, construindo confiança.*

**(TRANSPETRO – 2018)** Entre os processos de desenvolvimento de software ágeis mais usados no Brasil está o SCRUM. Quais são os pilares do SCRUM que apoiam a implementação de controle de processo empírico?

- a) Comprometimento, coragem, foco e respeito
- b) Comprometimento, transparência e adaptação
- c) Coragem, inspeção e adaptação
- d) Transparência, adaptação, foco e respeito
- e) Transparência, inspeção e adaptação

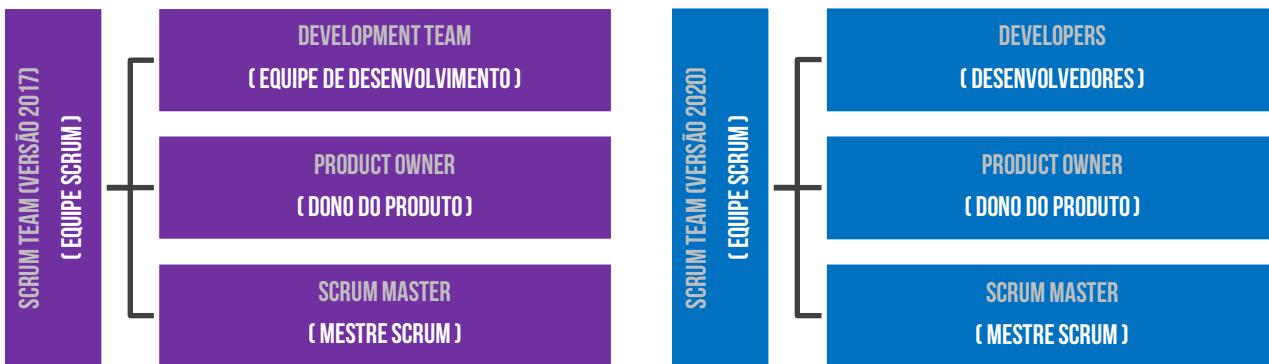
**Comentários:** a questão mistura valores (Foco, Abertura, Coragem, Comprometimento e Respeito) com pilares (Transparência, Inspeção e Adaptação) (Letra E).



## Papéis

INCIDÊNCIA EM PROVA: ALTÍSSIMA

O Scrum possui poucos papéis, mas muito bem definidos! As pessoas que desempenham esses papéis são igualmente responsáveis e responsabilizadas pelos resultados do trabalho e, assim, se comprometem com o projeto. Eles são membros de um mesmo time e trabalham juntos, de forma colaborativa, para alcançarem seus resultados. **Os papéis são diferentes dependendo da versão de referência:**



### [Guia Scrum - Versão 2017]

O Time Scrum consiste em um Product Owner, o Time de Desenvolvimento e um Scrum Master. Times Scrum são auto-organizáveis e multifuncionais. Times auto-organizáveis escolhem qual a melhor forma para completarem seu trabalho, em vez de serem dirigidos por outros de fora do Time. Times multifuncionais possuem todas as competências necessárias para completar o trabalho sem depender de outros que não fazem parte da equipe. O modelo de time no Scrum é projetado para aperfeiçoar a flexibilidade, criatividade e produtividade.

O Time Scrum demonstra-se estar aumentando sua efetividade para todos os usos anteriormente citados, e qualquer trabalho complexo. Times Scrum entregam produtos de forma iterativa e incremental, maximizando as oportunidades para feedback. Entregas incrementais de produto "Pronto" garantem que uma versão potencialmente funcional do produto do trabalho esteja sempre disponível. O tamanho ideal do Time de Desenvolvimento é pequeno o suficiente para se manter ágil e grande o suficiente para completar um trabalho significativo dentro da Sprint.

Menos de três integrantes no Time de Desenvolvimento diminuem a interação e resultam em um menor ganho de produtividade. Times de desenvolvimento menores podem encontrar restrições de habilidades durante a Sprint, gerando um Time de Desenvolvimento incapaz de entregar um incremento potencialmente liberável. Havendo mais de nove integrantes é exigida muita coordenação. Times de Desenvolvimento grandes geram muita complexidade para que um processo empírico seja útil. Os papéis de Product Owner e de Scrum Master não são incluídos nesta contagem, a menos que eles também executem o trabalho do Backlog da Sprint.

### [Guia Scrum - Versão 2020]

A unidade fundamental do Scrum é um pequeno time de pessoas, um Scrum Team. O Scrum Team consiste em um Scrum Master, um Product Owner e Developers. Dentro de um Scrum Team, não há sub-times ou hierarquias. É uma unidade coesa de profissionais focados em um objetivo de cada vez, a Meta do Produto. Os Scrum Teams 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 significa que decidem internamente quem faz o quê, quando e como.

O Scrum Team é pequeno o suficiente para permanecer ágil e grande o suficiente para concluir um trabalho significativo dentro de uma Sprint, normalmente 10 ou menos pessoas. Em geral, descobrimos que times menores se comunicam melhor e são mais produtivos. Se os Scrum Teams se tornarem muito grandes, eles devem considerar a reorganização em vários Scrum Teams coesos, cada um focado no mesmo produto. Portanto, eles devem compartilhar a mesma meta do produto, Product Backlog e Product Owner.

O Scrum Team é responsável por todas as atividades relacionadas ao produto, desde a colaboração com stakeholder, verificação, manutenção, operação, experimentação, pesquisa e desenvolvimento, e qualquer outra coisa que possa ser necessária. Eles são estruturados e empoderados pela organização para gerenciar seu próprio trabalho. Trabalhar em Sprints em um ritmo sustentável melhora o foco e a consistência do Scrum Team. Todo o Scrum Team é responsável por criar um Incremento valioso e útil a cada Sprint. Scrum define três responsabilidades específicas dentro do Scrum Team: os Developers, o Product Owner e o Scrum Master.

Na versão 2017, é importante não confundir Development Team com Scrum Team! **A Equipe Scrum que é um time auto-organizável e multifuncional!** Ser auto-organizável significa que ela é capaz de escolher qual a melhor forma para realizar seu próprio trabalho em vez de serem dirigidos por outros de fora do Time. Ser multifuncional significa que ela possui todas as competências e não depende de outros de fora da equipe.

Em outras palavras, os times de desenvolvimento não contêm subtimes dedicados a domínios específicos de conhecimento (Ex: testadores, analistas de negócio, entre outros). **A Equipe Scrum é o responsável por entregar produtos de forma iterativa e incremental, maximizando as oportunidades de realimentação (feedback).** Uma dúvida comum é: professor, pode existir uma sobreposição nesses papéis, isto é, uma mesma pessoa desempenhando dois papéis diferentes?

**Não! Scrum Master e o Product Owner podem fazer parte do Development Team [Versão 2017], mas um Scrum Master jamais pode ser simultaneamente Product Owner!** Essa última afirmação não está explícita no Guia Scrum, mas é possível inferir que pode haver um conflito de interesses. Quanto à primeira afirmação: "The Product Owner and Scrum Master roles are not included in this count unless they are also executing the work of the Sprint Backlog".

**(TJ-PE – 2017)** O Scrum está sendo implantado dentro da sua empresa, portanto existe a necessidade de se criar o Time Scrum que é formado pelo:

- a) Product Backlog, o Time de Planejamento e o Scrum Master
- b) Product Owner, o Time de Desenvolvimento e o Scrum Sprint
- c) Product Backlog, o Time de Planejamento e o Scrum Sprint
- d) Product Owner, o Time de Desenvolvimento e o Scrum Master
- e) Product Backlog, o Time de Desenvolvimento e o Scrum Sprint

**Comentários:** o Time Scrum é formado pelo Product Owner, Time de Desenvolvimento (Desenvolvedores) e Scrum Master (Letra D).



## Product Owner (PO)

O *Product Owner* é uma pessoa e, não, um comitê. **Ele pode representar o desejo de um comitê no Product Backlog, mas aqueles que quiserem uma alteração nas prioridades dos itens de backlog devem convencer o Product Owner.** Para que ele tenha sucesso, toda a organização deve respeitar as suas decisões e elas devem ser visíveis no conteúdo e na priorização do Backlog do Produto. Vejamos a seguir suas responsabilidades e características:

### RESPONSABILIDADES PRODUCT OWNER (PO)

- Ele é responsável pela macro-gestão e pela gestão do produto.
- Ele é o responsável por maximizar o valor do produto e do trabalho dos desenvolvedores, sendo o único que pode gerenciar o Product Backlog.
- Ele pode até delegar as atividades de gerenciamento para os desenvolvedores, mas ainda será considerado o responsável pelos trabalhos.
- Ele é responsável por priorizar/ordenar os itens do Product Backlog e seleciona aqueles que serão implementados.
- Ele é responsável por garantir o ROI (Retorno On Investment ou Retorno sobre Investimento).
- Ele é responsável por expressar claramente os itens do Product Backlog.
- Ele é responsável por garantir que o Backlog do Produto seja visível, transparente, claro para todos, e mostrar o que a Equipe Scrum vai trabalhar a seguir.
- Ele é responsável por garantir que os desenvolvedores entendam os itens do Product Backlog no nível necessário.

### [Guia Scrum - Versão 2017]

*O Product Owner, ou dono do produto, é o responsável por maximizar o valor do produto resultado do trabalho do Time de Desenvolvimento. Como isso é feito pode variar amplamente através das organizações, Times Scrum e indivíduos. O Product Owner é a única pessoa responsável por gerenciar o Backlog do Produto. O gerenciamento do Backlog do Produto inclui:*

- Expressar claramente os itens do Backlog do Produto;
- Ordenar os itens do Backlog do Produto para alcançar melhor as metas e missões;
- Otimizar o valor do trabalho que o Time de Desenvolvimento realiza;
- Garantir que o Backlog do Produto seja visível, transparente, claro para todos, e mostrar o que o Time Scrum vai trabalhar a seguir; e,
- Garantir que o Time de Desenvolvimento entenda os itens do Backlog do Produto no nível necessário.

*O Product Owner pode fazer o trabalho acima, ou delegar para o Time de Desenvolvimento fazê-lo. No entanto, o Product Owner continua sendo o responsável pelos trabalhos. O Product Owner é uma pessoa e não um comitê. O Product Owner pode representar o desejo de um comitê no Backlog do Produto, mas aqueles que quiserem uma alteração nas prioridades dos itens de Backlog devem endereçar ao Product Owner. Para que o Product Owner tenha sucesso, toda a organização deve respeitar as decisões dele(a). As decisões do Product Owner são visíveis no conteúdo e na priorização do Backlog do Produto. Ninguém pode forçar o Time de Desenvolvimento a trabalhar em um diferente conjunto de requerimentos.*

### [Guia Scrum - Versão 2020]



O Product Owner 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, Scrum Teams e indivíduos. O Product Owner também é responsável pelo gerenciamento eficaz do Product Backlog, que inclui:

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

O Product Owner pode fazer o trabalho acima ou pode delegar a responsabilidade a outros. Independentemente disso, o Product Owner ainda é o responsável. Para que os Product Owners tenham sucesso, toda a organização deve respeitar suas decisões. Essas decisões são visíveis no conteúdo e na ordem do Product Backlog e por meio do incremento inspecionável na revisão da sprint. O Product Owner é uma pessoa, não um comitê. O Product Owner pode representar as necessidades de muitos stakeholders no Product Backlog. Aqueles que desejam alterar o Product Backlog podem fazê-lo tentando convencer o Product Owner.

**(DETRAN-MT – 2015)** No Scrum, há três papéis importantes: Product Owner, Team e Scrum Master. É responsabilidade do Product Owner:

- Determinar como serão a gestão e a organização dos times.
- Desenvolver funcionalidades do projeto.
- Assegurar que a funcionalidade mais valiosa será produzida primeiro.
- Ensinar Scrum a todos os envolvidos no projeto.

**Comentários:** (a) Errado, essa não é uma responsabilidade do Product Owner; (b) Errado, essa é uma responsabilidade dos desenvolvedores; (c) Correto, ele é responsável por priorizar as funcionalidades no Product Backlog; (d) Errado, essa é uma responsabilidade do Scrum Master (Letra C).

**(UFRJ – 2018)** No framework Scrum, o papel que tem como uma das responsabilidades, maximizar o valor do produto e do trabalho do Time de Desenvolvimento, além de ser a pessoa responsável por gerenciar o Backlog do Produto é denominado:

- Product Owner.
- Scrum Master.
- Scrum Team.
- Stakeholder.
- Development Team.

**Comentários:** o papel responsável por maximizar o valor do produto e do trabalho do time de desenvolvimento (desenvolvedores), além de ser a pessoa responsável por gerenciar o Backlog do Produto é o Product Owner (Letra A).



## Developers (DV)

**Os Desenvolvedores consistem em profissionais que realizam o trabalho de entregar uma versão usável que potencialmente incrementa o produto “pronto” ao final de cada sprint.** Somente Desenvolvedores criam incrementos. Ninguém tem permissão para falar com os eles sobre diferentes configurações de prioridade e os eles não têm permissão para agir sobre o que outras pessoas disserem.

**Os Desenvolvedores só respondem ao Product Owner. Além disso, só ele pode cancelar uma sprint.** Ele deve ser pequeno o suficiente de forma a se manter ágil e produtivo, e grande o suficiente de forma que a coordenação dos membros não cause problemas. A versão anterior do guia recomendava que a equipe tivesse entre 3 e 9 integrantes (excluídos o Product Owner e Scrum Master – exceto se eles também executarem o trabalho da sprint).

No Guia Scrum 2020, houve uma alteração de nomes. O termo ‘Time de Desenvolvimento’ passava a impressão de que existia um sub-time no Time Scrum. Esse termo foi substituído por uma nova responsabilidade: Desenvolvedores. Isso reforça a ideia de existir apenas um único time: o Time Scrum – que é formado pelo Scrum Master, o Product Owner e os Desenvolvedores, sendo que seus objetivos devem ser os mesmos.

A ideia por trás dessa mudança não é apenas semântica: ao remover o conceito de sub-time dentro da equipe e deixar claro que todas essas pessoas pertencem ao mesmo time, o Time Scrum, isso cria um compromisso mais forte entre todos para a entrega da Meta da Sprint. O Time Scrum é, portanto, apenas uma equipe com três responsabilidades diferentes e objetivos idênticos. *Fechado?*

A versão anterior dizia que o Time de Desenvolvimento deveria idealmente ser composto por 3 a 9 integrantes. Com o objetivo de se tornar ainda menos prescritivo, agora não há tamanho mínimo ou máximo para os desenvolvedores. No entanto, o Scrum Guide comenta que o Scrum Team normalmente tem 10 ou menos integrantes – incluindo na conta o Scrum Master e o Product Owner. Não mudou muita coisa da versão anterior, mas deixou mais aberto...

RESPONSABILIDADES E  
CARACTERÍSTICAS: DESENVOLVEDORES

Responsável pela micro-gestão e pela criação do produto.

Eles são auto-organizados. Ninguém (nem mesmo o SM) diz aos desenvolvedores como transformar o Product Backlog em incrementos de funcionalidades potencialmente utilizáveis.

Times de Desenvolvimento são multifuncionais, possuindo todas as habilidades necessárias, enquanto equipe, para criar o incremento do Produto.

O Scrum não reconhece títulos específicos para os desenvolvedores, independentemente do trabalho que está sendo realizado pela pessoa;

Individualmente, os desenvolvedores podem ter habilidades especializadas, mas a responsabilidade pertence aos desenvolvedores como um todo.

Os desenvolvedores não contêm sub-times dedicados a domínios específicos de conhecimento, tais como teste ou análise de negócios.



Os desenvolvedores são estruturados e autorizados pela organização para organizar e gerenciar seu próprio trabalho.

**[Guia Scrum - Versão 2017]**

O Time de Desenvolvimento consiste de profissionais que realizam o trabalho de entregar um incremento potencialmente liberável do produto "Pronto" ao final de cada Sprint. Um incremento "Pronto" é requerido na Revisão da Sprint. Somente integrantes do Time de Desenvolvimento criam incrementos. Os Times de Desenvolvimento são estruturados e autorizados pela organização para organizar e gerenciar seu próprio trabalho. A sinergia resultante aperfeiçoa a eficiência e a eficácia do Time de Desenvolvimento como um todo.

**[Guia Scrum - Versão 2020]**

Developers são as pessoas do Scrum Team que estão comprometidas em criar qualquer aspecto de um Incremento utilizável a cada Sprint. As habilidades específicas necessárias pelos Developers geralmente são amplas e variam de acordo com o domínio de trabalho. No entanto, os Developers são sempre responsáveis por: criar um plano para a Sprint, o Sprint Backlog; introduzir gradualmente qualidade aderindo a uma Definição de Pronto; adaptar seu plano a cada dia em direção à meta da Sprint; e responsabilizar-se mutuamente como profissionais.

**(Banco da Amazônia – 2018)** No SCRUM, o Backlog da Sprint é “um conjunto de itens do Backlog do Produto selecionados para Sprint, juntamente com o plano para entregar o incremento do produto e atingir o objetivo da Sprint”

(Schwaber e Sutherland, 2017).

Durante a Sprint, quem pode alterar o Backlog da Sprint?

- a) Product Owner, apenas
- b) Scrum Master, apenas
- c) Time de Desenvolvimento, apenas
- d) Time de Desenvolvimento e o Product Owner, apenas
- e) Time de Desenvolvimento e o Scrum Master, apenas

**Comentários:** durante a sprint, apenas o Time de Desenvolvimento (Desenvolvedores) pode alterar o Sprint Backlog (Letra C).

**(TRE-MT – 2015)** Em um projeto ágil em que se utiliza Scrum, a criação e a estimação de tarefas cabe ao:

- a) gerente de projeto.
- b) product owner.
- c) time de desenvolvimento.
- d) Scrum master.
- e) time de desenvolvimento, ao Scrum master e ao product owner, em conjunto.

**Comentários:** a criação e a estimação de tarefas cabem ao Time de Desenvolvimento ou Desenvolvedores (Letra C).



## Scrum Master (SM)

O **Scrum Master** é responsável por garantir que o Scrum seja entendido e aplicado! Ele faz isso para garantir que a Equipe Scrum adira à teoria, práticas e regras do Scrum. O Scrum Master é um servo-líder para a Equipe Scrum. Ele ajuda aqueles que estão fora da Equipe Scrum a entender quais as suas interações com a Equipe Scrum são úteis e quais não são. Ademais, ele ajuda todos a mudarem estas interações para maximizar o valor criado pela Equipe Scrum.

Responsável pela gestão de pessoas e gestão do processo.

Ele deve garantir que o Scrum seja entendido e aplicado. O Scrum Master faz isso para garantir que a Equipe Scrum adere à teoria, práticas e regras do Scrum.

O Scrum Master ajuda aqueles que estão fora da Equipe Scrum a entender quais as suas interações com a Equipe Scrum são úteis e quais não são.

O Scrum Master ajuda todos a mudarem estas interações para maximizar o valor criado pela Equipe Scrum.

Ele é responsável por orientar o Product Owner na criação e ordenação do Product Backlog.

Ele é responsável por garantir que as regras do Scrum estejam sendo cumpridas e seus valores estejam sendo seguidos.

Ele é responsável por ajudar a remover impedimentos que o time encontre, fazendo isso sem o uso de qualquer autoridade.

Ele utiliza técnicas de facilitação e coaching para que os membros do time consigam visualizar os problemas e encontrem a melhor solução.

Durante eventos, ele é responsável por fazer com que a reunião flua adequadamente, utilizando técnicas de facilitação, embora não seja o responsável pela condução.

Ele ajuda a treinar os desenvolvedores em autogerenciamento e interdisciplinaridade.

Ele treina os desenvolvedores em ambientes organizacionais nos quais o Scrum não é totalmente adotado e compreendido.

Ele ensina a Equipe Scrum a criar itens do Product Backlog de forma clara e concisa.

Ele comunica claramente a visão, objetivo e itens do Product Backlog para os desenvolvedores.

### RESPONSABILIDADES SCRUM MASTER (SM)

#### [Guia Scrum - Versão 2017]

O Scrum Master é responsável por promover e suportar o Scrum como definido no Guia Scrum. O Scrum Master faz isso ajudando todos a entenderem a teoria, as práticas, as regras e os valores do Scrum. O Scrum Master é um servo-líder para o Time Scrum. O Scrum Master ajuda aqueles que estão fora do Time Scrum a entender quais as suas interações com o Time Scrum são úteis e quais não são. O Scrum Master ajuda todos a mudarem estas interações para maximizar o valor criado pelo Time Scrum.

#### O Scrum Master trabalhando para o Product Owner

O Scrum Master serve o Product Owner de várias maneiras, incluindo:



- Garantindo que objetivos, escopo e domínio do produto sejam entendidos o melhor possível por todos do Time Scrum;
- Encontrando técnicas para o gerenciamento efetivo do Backlog do Produto;
- Ajudando o Time Scrum a entender as necessidades para ter itens de Backlog do Produto claros e concisos.
- Compreendendo o planejamento do Produto em um ambiente empírico;
- Garantindo que o Product Owner saiba como organizar o Backlog do Produto para maximizar valor;
- Compreender e praticar a agilidade; e,
- Facilitar os eventos Scrum conforme exigidos ou necessários.

### **O Scrum Master trabalhando para o Time de Desenvolvimento**

O Scrum Master serve o Time de Desenvolvimento de várias maneiras, incluindo:

- Treinando o Time de Desenvolvimento em autogerenciamento e interdisciplinaridade;
- Ajudando o Time de Desenvolvimento na criação de produtos de alto valor;
- Removendo impedimentos para o progresso do Time de Desenvolvimento;
- Facilitando os eventos Scrum conforme exigidos ou necessários; e,
- Treinando o Time de Desenvolvimento em ambientes organizacionais nos quais o Scrum não é totalmente adotado e compreendido.

### **O Scrum Master trabalhando para a Organização**

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

- Liderando e treinando a organização na adoção do Scrum;
- Planejando implementações Scrum dentro da organização;
- Ajudando funcionários e partes interessadas a compreender e tornar aplicável o Scrum e o desenvolvimento de produto empírico;
- Causando mudanças que aumentam a produtividade do Time Scrum; e,
- Trabalhando com outros Scrum Masters para aumentar a eficácia da aplicação do Scrum na organização.

### **[Guia Scrum - Versão 2020]**

O Scrum Master é responsável por estabelecer o Scrum conforme definido no Guia do Scrum. Eles fazem isso ajudando todos a entender a teoria e a prática do Scrum, tanto no Scrum Team quanto na organização. O Scrum Master é responsável pela eficácia do Scrum Team. Eles fazem isso permitindo que o Scrum Team melhore suas práticas, dentro do framework Scrum. Scrum Masters são verdadeiros líderes que servem ao Scrum Team e à organização como um todo.

### **O Scrum Master serve ao Scrum Team 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 atendem à Definição de Pronto;
- Provocando a remoção de impedimentos ao progresso do Scrum Team; e,
- Garantir que todos os eventos Scrum ocorram e sejam positivos, produtivos e mantidos dentro do Timebox.

### **O Scrum Master serve o Product Owner 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; e,
- Facilitar a colaboração dos stakeholder, conforme solicitado ou necessário.



O Scrum Master serve a 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 compreender e aplicar uma abordagem empírica para trabalhos complexos; e,
- Remover barreiras entre stakeholders e Scrum Teams.

Galera, eu só consigo explicar por metáfora, então vamos tentar entender esses papéis! Imaginem que João deseja construir uma casa. Para tal, ele contrata uma renomada empresa de engenharia. A empresa irá fornecer todo seu know-how por meio de uma Equipe de Construção de Casas, **que será composta por uma Equipe de Pedreiros, um Mestre de Obras e... por você!** Sim, você fará parte da Equipe de Construção de Casas como principal parte interessada.

*Vamos dar um nome para você?* Você ocupará o cargo de Dono da Casa. Portanto, a Equipe de Construção de Casas será composta por uma Equipe de Pedreiros, pelo Mestre de Obras e pelo Dono da Casa. *E como será a organização e a função de cada um desses papéis?* **Bem, a Equipe de Pedreiros é composta por 3 a 9 pedreiros multidisciplinares, isto é, todos dominam todas as atividades de um pedreiro.**

Galera, esses caras são os responsáveis por colocar a mão na massa, levantar parede, fazer o concreto, alinhar o piso, entre outras atividades. Já o Mestre de Obras é o grande facilitador! *Como assim, professor?* **Ele é o cara que vai retirar os impedimentos que aparecem no decorrer do nosso projeto.** Um pedreiro faltou? Ficou doente? Se machucou? Ele irá buscar uma maneira de reduzir os impactos dessa ausência.

*Os pedreiros estão desmotivados, distraídos, descuidados?* Ele irá arrumar uma maneira de solucionar isso. *Um pedreiro saiu na porrada com outro?* Ele vai tentar intermediar o conflito. **Além disso, ele que vai trazer a demanda do Dono da Casa, entender o que ele quer e passar de maneira simples, clara e objetiva para a Equipe de Pedreiros.** Ele é o cara que mais deve conhecer os fundamentos da construção de uma casa!

Imaginem ele como um grande estudioso da construção de casas com bastante experiência adquirida em anos de empreendimentos. Ele saca tudo sobre como se faz para se construir uma casa, qual é o papel de cada um, qual é a melhor ordem, quais são as melhores técnicas e é um grande professor, capaz de ensinar a todos os outros integrantes quais são os melhores princípios para a construção de casas. **Em outras palavras, ele é um grande mestre!**

Por fim, ele também é capaz de treinar a equipe de construção para que ela seja auto-gerenciável e interdisciplinar. Legal, mas está faltando um papel nessa história. *E qual é?* É o seu papel! **Qual a sua função?** Você é o Dono da Casa! **Logo, você que gerencia o que será feito e o que não será feito, sendo também o responsável pelo que será entregue.** A Equipe de Pedreiros responde a você! Até o Mestre de Obras responde somente a você! *Viu a sua importância?*



Você é o cara que tem que tentar fazer essa casa ficar da melhor forma possível – com máximo valor agregado. **Você é o cara que vai priorizar o que deve ser feito primeiro; você é o cara que coloca ou tira tarefas da lista tarefas a serem realizadas;** você é o único cara que pode cancelar determinada tarefa; você é o cara que fiz expressamente o que deve ser feito; enfim... você ocupa um papel importantíssimo para o sucesso do projeto de construir uma casa.

Galera, toda metáfora possui suas limitações, mas acho que vocês conseguiram captar a mensagem aqui! A Equipe de Construção de Casas é o *Scrum Team*; a Equipe de Pedreiros é o *Development Team*; o Mestre de Obras é o *Scrum Master*; e o Dono da Casa é o *Product Owner*. **Além disso, cada um desses tem atividades bem definidas e o controle sobre essas atividades é descentralizado.** Entendido? Seeeeegue o jogo...

**(TRE-RS – 2015)** No desenvolvimento ágil de sistemas utilizando o Scrum, um integrante da equipe é encarregado de comunicar a visão, os objetivos e os itens do product backlog para o time de desenvolvimento, além de encontrar técnicas para o gerenciamento efetivo do product backlog. Esse integrante é o:

- a) Product Owner, sob orientação do Scrum Master.
- b) próprio time de desenvolvimento, que realiza essas definições de forma auto-organizada.
- c) Scrum Master.
- d) Team Leader.
- e) Product Owner, diretamente.

**Comentários:** o integrante encarregado de comunicar a visão, os objetivos e os itens do Product Backlog para o time de desenvolvimento, além de encontrar técnicas para o gerenciamento efetivo do Product Backlog é o Scrum Master (Letra C).

**(EPTC – 2012)** No Scrum, existem papéis bem definidos. Assinalar a alternativa a qual o trecho abaixo se refere:

*Tem como função primária remover impedimentos para que a equipe consiga entregar o objetivo do Sprint. Além dessa função, a pessoa nesse papel tem a função de assegurar que as práticas do Scrum sejam utilizadas corretamente.*

- a) Scrum Product Owner.
- b) Scrum Master.
- c) Scrum Manager.
- d) Scrum Project Manager.

**Comentários:** esse é o papel do Scrum Master (Letra B).



## Artefatos

INCIDÊNCIA EM PROVA: ALTÍSSIMA

Segundo o Guia do Scrum, o framework possui apenas três artefatos oficiais: *Product Backlog*, *Sprint Backlog* e *Product Increment*. Um artefato é o produto de um trabalho...

### [Guia Scrum - Versão 2017]

*Os artefatos do Scrum representam o trabalho ou o valor para o fornecimento de transparência e oportunidades para inspeção e adaptação. Os artefatos definidos para o Scrum são especificamente projetados para maximizar a transparência das informações chave de modo que todos tenham o mesmo entendimento dos artefatos*

### [Guia Scrum - Versão 2020]

*Os artefatos do Scrum representam trabalho ou valor. Eles são projetados para maximizar a transparência das principais informações. Assim, todos os que os inspecionam têm a mesma base para adaptação. Cada artefato contém um compromisso para garantir que ele forneça informações que aumentem a transparência e o foco contra o qual o progresso pode ser medido:*

- Para o *Product Backlog*, é a *Meta do produto*.
- Para o *Sprint Backlog*, é a *Meta da Sprint*.
- Para o *incremento*, é a *Definição de Pronto*.

*Esses compromissos existem para reforçar o empirismo e os valores Scrum para o Scrum Team, e seus stakeholders.*

## Product Backlog

### PRODUCT BACKLOG

Trata-se de uma lista ordenada (por valor, risco, prioridade, entre outros) de requisitos ou funcionalidades que o produto deve conter criada pela Equipe Scrum e gerenciada pelo Product Owner.

*Antes de tudo, o que é um backlog?* Galera, um *backlog* é basicamente uma lista, um resumo histórico, de acumulação de trabalho num determinado período de tempo, pode ser uma pilha de pedidos que devem ser produzidos. **Já o *Product Backlog* é uma lista ordenada (por valor, risco, prioridade, entre outros) de requisitos ou funcionalidades que o produto deve conter<sup>1</sup> criada pela Equipe Scrum.**

O *Product Backlog* é a origem única dos requisitos para qualquer mudança a ser feita no produto. **Costuma-se dizer que ele é um artefato dinâmico que nunca estará completo e existirá enquanto o produto também existir. Por que?** Porque sempre haverá novos requisitos, novas

<sup>1</sup> Apesar de o Scrum Team (PO, SM, DV) ser o responsável pela criação do *Product Backlog*, o responsável pelo artefato e o único que pode priorizar suas funcionalidades é o *Product Owner*.



necessidades e mudanças a serem incorporadas. Logo, trata-se de um artefato vivo – sempre em movimento.

**O Product Backlog evolui tanto quanto o produto e o ambiente no qual ele será utilizado evoluem.** Ele muda constantemente para identificar o que o produto necessita para ser mais apropriado, competitivo e útil para as partes interessadas. Lembrando que somente o *Product Owner* pode inserir, remover ou reordenar esses itens, incluindo seu conteúdo, disponibilidade e ordenação.

Rafael Prikladnicki afirma que o formato mais utilizado para os itens são Histórias de Usuário (*User Stories*) ordenadas de acordo com o critério escolhido pelo *Product Owner*. *O que são histórias de usuário?* É basicamente uma especificação de uma ou mais frases na linguagem de negócios ou cotidiana do usuário do sistema que captura o que um usuário faz ou necessita fazer como parte de sua função de trabalho.

**Enfim... em geral, itens mais importantes ficam no topo do Product Backlog e são implementados primeiro.** Na maioria das vezes, esses são os itens sobre os quais há maior conhecimento, logo são mais detalhados e refinados. Itens que precisem de maior refinamento geralmente têm uma importância menor e ficam mais abaixo. Não existe, no entanto, uma forma única para materializar esse artefato e descrever seus itens.

Além das Histórias de Usuário, podem ser utilizadas descrições textuais de funcionalidades, cenários de casos de uso, etc. Galera, o *Product Backlog* pode apresentar itens funcionais, não-funcionais, arquiteturais ou de infraestrutura – **além de itens que representem riscos a serem removidos.** É claro que, durante o andamento do projeto, algumas funcionalidades podem acabar perdendo a importância – não importando sob que circunstâncias isso ocorreu.

Backlog do Produto: Twitter
Login de usuários já cadastrados
Cadastrar novo usuário
Tuitar
Visualizar tuítes de quem sigo
Visualizar número de seguidores que posso
Visualizar histórico de um tuíte
Mostrar banner promocional
Remover tuíte
Ver sugestões de usuários a seguir
Listar trends
Compor e alterar meu perfil de usuário
(...)

**Isso é totalmente normal na maioria dos projetos, uma vez que é impossível saber, desde o início, os detalhes de tudo o que queremos no produto.** Assim, algumas funcionalidades podem acabar até mesmo desaparecendo. Da mesma forma, novas funcionalidades também podem ser adicionadas de acordo com a necessidade. Ao lado, temos um exemplo de *Product Backlog*. É nele onde eu vou armazenar todas as necessidades que eu desejo no meu projeto, entre outras coisas. No exemplo acima, deseja-se poder tanto tuitar quanto remover um tuíte – são duas histórias de usuário diferentes que eu desejo que sejam implementadas na minha aplicação. *Legal, mas quando eu sei quando um desses itens pode realmente ser considerado pronto (também chamado ready)?*



Para que um item possa ser incluído em uma sprint, ele deve ser pequeno o suficiente para que caiba em uma única sprint e deve deixar tudo claro e transparente quanto à expectativa do *Product Owner* (geralmente através de um critério de aceite). *Mas até que ponto estes requisitos precisam ser detalhados? Eles devem ser detalhados até atender ao conceito de Definition of Ready (DoR). O que é isso, Diego?*

**Isso significa que o requisito tem informações suficientes para começar a ser desenvolvido imediatamente.** Esta definição é bastante específica de cada organização – não há um padrão. Vou dar um exemplo para melhor entendimento... eu já trabalhei em um projeto em que as histórias de usuário eram entregues aos desenvolvedores de forma muito pobres, pouco refinadas e demasiadamente confusas.

**O Product Owner estava rejeitando a conclusão das sprints, declarando que não havia sido feito o que ele havia pedido.** Os desenvolvedores reclamavam que a especificação era péssima e que nem o próprio *Product Owner* sabia o que queria. A partir daí, modificamos nosso processo! A equipe combinou critérios explícitos e visíveis do que uma história de usuário deveria conter para ser aceita para entrar em uma sprint. Como diz o ditado, combinado não sai caro!

Dessa forma, uma vez que todos haviam concordado, as brigas reduziram bastante. *Por que?* Porque eles nos disseram exatamente (por meio de um checklist) **o que a história de usuário deveria conter para que elas pudessem ser aceitas para entrar no Product Backlog e serem de fato implementadas pelos desenvolvedores**. A partir daí, essa definição de “pronto” nos ajudou a mitigar falhas de comunicação.

#### [Guia Scrum - Versão 2017]

*O Backlog do Produto é uma lista ordenada de tudo que é conhecido ser necessário no produto. É a única origem dos requisitos para qualquer mudança a ser feita no produto. O Product Owner é responsável pelo Backlog do Produto, incluindo seu conteúdo, disponibilidade e ordenação. Um Backlog do Produto nunca está completo. Os primeiros desenvolvimentos estabelecem os requisitos inicialmente conhecidos e melhor entendidos. O Backlog do Produto evolui tanto quanto o produto e o ambiente no qual ele será utilizado evoluem.*

*O Backlog do Produto é dinâmico; mudando constantemente para identificar o que o produto necessita para ser mais apropriado, competitivo e útil. Se um produto existe, seu Backlog do Produto também existe. O Backlog do Produto lista todas as características, funções, requisitos, melhorias e correções que formam as mudanças que devem ser feitas no produto nas futuras versões. Os itens do Backlog do Produto possuem os atributos de descrição, ordem, estimativa e valor. Os itens do Backlog geralmente incluem descrições de testes que comprovarão sua completude quando “Prontos”.*

*Enquanto um produto é usado e ganha valor, e o mercado fornece feedback, o Backlog do Produto torna-se uma lista maior e mais completa. Requisitos nunca param de mudar, então o Backlog do Produto é um artefato vivo. Mudanças nos requisitos de negócio, condições de mercado ou tecnologia podem causar mudanças no Backlog do Produto. Múltiplos Times Scrum frequentemente trabalham juntos no mesmo produto. Um Backlog do Produto é usado para descrever o trabalho previsto para o produto. Um atributo do Backlog do Produto que agrupe itens pode ser então aplicado.*

*O refinamento do Backlog do Produto é a ação de adicionar detalhes, estimativas e ordem aos itens no Backlog do Produto. Este é um processo contínuo no qual o Product Owner e o Time de Desenvolvimento colaboram nos detalhes dos itens do Backlog do Produto. Durante o refinamento do Backlog do Produto, os itens são inspecionados e revisados. O*



Time de Scrum decide como e quando o refinamento está "Pronto". Este refinamento usualmente não consome mais de 10% da capacidade do Time de Desenvolvimento.

Contudo, os itens do Backlog do Produto podem ser atualizados a qualquer momento pelo Product Owner ou a critério do Product Owner. Os itens do Backlog do Produto de ordem mais alta (topo da lista) devem ser mais claros e mais detalhados que os itens de ordem mais baixa. Estimativas mais precisas são feitas baseadas em maior clareza e maior detalhamento; Quanto menor a ordem na lista, menos detalhes. Os itens do Backlog do Produto que irão ocupar o Time de Desenvolvimento na próxima Sprint são mais refinados, de modo que todos os itens possam ser "Prontos" dentro do time-boxed da Sprint.

Os itens do Backlog do Produto que podem ser "Prontos" pelo Time de Desenvolvimento dentro de uma Sprint são considerados "Preparados" para seleção no Planejamento da Sprint. Itens do Backlog do Produto geralmente adquirem este grau de transparência através das atividades de refinamento descritas acima. O Time de Desenvolvimento é responsável por todas as estimativas. O Product Owner deve influenciar o Time de Desenvolvimento, ajudando no entendimento e nas decisões conflituosas de troca, mas as pessoas que irão realizar o trabalho fazem a estimativa final.

### **Monitorando o Progresso a Caminho dos Objetivos**

Em qualquer ponto do tempo, o total do trabalho restante para alcançar o objetivo pode ser somado. O Product Owner acompanha o total do trabalho restante pelo menos a cada Revisão da Sprint. O Product Owner compara este valor com o trabalho restante nas Revisões das Sprints anteriores, para avaliar o progresso na direção de completar o trabalho previsto pelo tempo desejado para alcançar o objetivo.

Esta informação deve ser transparente para todas as partes interessadas. Várias práticas para prever tendências foram usadas para prever o progresso, tais como burndowns, burn-ups, ou fluxos cumulativos. Estas têm se provado úteis. Contudo, não substituem a importância do empirismo. Em ambientes complexos, o que acontecerá é desconhecido. Somente o que já ocorreu pode ser usado para uma tomada de decisão a respeito do que virá.

### **[Guia Scrum - Versão 2020]**

O Product Backlog é uma lista ordenada e emergente do que é necessário para melhorar o produto. É a única fonte de trabalho realizado pelo Scrum Team. Os itens do Product Backlog que podem ser realizados pelo Scrum Team em uma Sprint são considerados preparados para seleção no evento Sprint Planning. Eles geralmente adquirem esse grau de transparência após as atividades de refinamento. O Product Backlog refinement é o ato de quebrar e incluir definição adicional aos itens do Product Backlog para ter itens menores e mais precisos.

Esta é uma atividade contínua para adicionar detalhes, como descrição, ordem e tamanho. Os atributos geralmente variam de acordo com o domínio de trabalho. Os Developers que farão o trabalho são responsáveis pelo dimensionamento. O Product Owner pode influenciar os Developers, ajudando-os a entender e selecionar trade-offs (trocas de itens).

### **Compromisso: Meta do Produto**

A Meta do Produto descreve um estado futuro do produto que pode servir como um alvo para o Scrum Team planejar. A Meta do produto está no Product Backlog. O restante do Product Backlog emerge para definir "o que" cumprirá a Meta do Produto. Um produto é um veículo para entregar valor. Tem um limite claro, stakeholders conhecidos, usuários ou clientes bem definidos. Um produto pode ser um serviço, um produto físico ou algo mais abstrato. A Meta do Produto é o objetivo de longo prazo para o Scrum Team. Eles devem cumprir (ou abandonar) um objetivo antes de assumir o próximo.



**(TRT7 – 2017)** Assinale a opção que apresenta o termo no qual constam as solicitações de melhorias e novas funcionalidades do software no método Scrum:

- a) Sprint Backlog
- b) Daily Scrum
- c) Sprint Planning
- d) Product Backlog

**Comentários:** as solicitações de melhorias e novas funcionalidades ficam no Product Backlog (Letra D).

**(CREA-AC – 2016)** Uma equipe de desenvolvimento está utilizando o SCRUM como modelo de desenvolvimento ágil. Nesse caso, o componente desse modelo que representa a visão geral do produto, definindo o que deve ser feito, assim como suas prioridades e a ordem em que deve ser realizado, é o:

- a) evaluation screen.
- b) given pillar.
- c) product backlog.
- d) sprint backlog.
- e) users visions.

**Comentários:** o componente que define o que deve ser feito, assim como ordem e prioridade é o Product Backlog (Letra C).

**(TRT3 – 2015)** Um técnico de TI está trabalhando em um projeto de desenvolvimento de software que utiliza o modelo Scrum, em que as funcionalidades a serem implementadas, na forma de histórias de usuários, são mantidas em uma lista denominada:

- a) product backlog.
- b) sprint.
- c) chaos list.
- d) sprint burndown.
- e) metaphor list.

**Comentários:** as funcionalidades a serem implementadas, na forma de histórias de usuários, são mantidas em uma lista ordenada chamada de Product Backlog (Letra A).



## Sprint Backlog

### SPRINT BACKLOG

Trata-se de conjunto de itens selecionados do Product Backlog, mas a meta da sprint e mais um plano de ação para entregar um incremento potencialmente usável – é criado e gerenciado pelos desenvolvedores.

O **Sprint Backlog** é o conjunto de itens selecionados para serem implementados durante a sprint mais o plano para transformá-los em um incremento. Assim, ao final de cada Reunião de Planejamento, um novo *Sprint Backlog* é criado. Normalmente, o plano é composto por tarefas técnicas necessárias para transformar o item em um incremento do produto. *Vamos diferenciar Product Backlog e Sprint Backlog?* Essa é uma pegadinha comum em prova!

O primeiro é uma lista ordenada dos requisitos ou funcionalidades que o software deverá possuir. O segundo é uma lista de tarefas a serem executadas durante uma sprint para atingir a sua meta. Trata-se do desmembramento de cada item selecionado do *Product Backlog* em pequenas tarefas. O *Sprint Backlog* torna visível todo o trabalho que os desenvolvedores identificam como necessário para atingir a meta da sprint.

Aliás, os desenvolvedores (e somente eles) podem adicionar novas tarefas caso descubram, no decorrer da sprint, que mais trabalho será necessário. Da mesma forma, também podem remover tarefas caso estas se mostrem desnecessárias. Conforme o trabalho é realizado ou completado, a estimativa do trabalho restante é atualizada. **Em qualquer ponto do tempo na sprint, o total do trabalho remanescente dos itens pode ser somado.**



O Sprint Backlog é altamente visível, uma imagem em tempo real do trabalho que os desenvolvedores planejam completar durante a sprint, e pertence exclusivamente os desenvolvedores. Eles monitoram o total do trabalho restante pelo menos a cada Reunião Diária. **Os desenvolvedores acompanham estes resumos diários e projeta a probabilidade de alcançar o objetivo da sprint.** Com o rastreamento do trabalho restante em toda a sprint, os desenvolvedores são capazes de gerenciar o seu progresso.

Na Versão 2017, foi incluído um texto que reforça muito a importância da melhoria contínua ao longo dos trabalhos do Time Scrum. Passa a ser declaradamente fundamental a inserção de pelo menos um item priorizado referente a melhoria do processo identificado na última reunião de retrospectiva. Essa alteração reforça explicitamente que o time deve trabalhar também na melhoria contínua do próprio time, trabalho e processos. Esse trecho foi retirado na Versão 2020.

Vejam a seguir um exemplo de Backlog da Sprint. No exemplo, a meta da Sprint 2 é que usuários possam estar no twitter. À esquerda, temos as funcionalidades do Backlog do Produto; e à direita, temos as tarefas ou atividades técnicas e de negócio do Backlog da Sprint que devem ser



executadas para entregar as funcionalidades e efetivamente cumprir a meta da sprint definida na reunião de planejamento.

Backlog da Sprint #2		
META DA SPRINT: USUÁRIOS PODERÃO ESTAR NO TWITTER		
Login de usuários já cadastrados	Ativar login com usuário GMail	Montar layout do box de login
	Montar plano de segurança	Testar integrado
	Estruturar log	Revisar código
	Criar comportamento de login	Inserir hint explicativo de funcionamento
	Atualizar documentação técnica	(...)
Cadastrar novo usuário	Criar tabelas no banco de dados	Estruturar persistência
	Definição sobre uso de templates	Quando validado, ativar usuário
	Escrever testes	Definir padrões para cadastros
	Atualizar documentação técnica	Validar e-mail cadastrado
	Testar integrado	(...)

#### [Guia Scrum - Versão 2017]

O Backlog da Sprint é um conjunto de itens do Backlog do Produto selecionados para a Sprint, juntamente com o plano para entregar o incremento do produto e atingir o objetivo da Sprint. O Backlog da Sprint é a previsão do Time de Desenvolvimento sobre qual funcionalidade estará no próximo incremento e sobre o trabalho necessário para entregar essa funcionalidade em um incremento "Pronto". O Backlog da Sprint torna visível todo o trabalho que o Time de Desenvolvimento identifica como necessário para atingir o objetivo da Sprint.

Para garantir melhoria contínua, é incluído no mínimo um item de prioridade alta sobre melhoria do processo identificado na última Reunião de Retrospectiva. O Backlog da Sprint é um plano com detalhes suficientes que as mudanças no progresso sejam entendidas durante a Reunião Diária. O Time de Desenvolvimento modifica o Backlog da Sprint ao longo de toda a Sprint, e o Backlog da Sprint vai surgindo durante a Sprint. Este surgimento ocorre quando o Time de Desenvolvimento trabalha segundo o plano e aprende mais sobre o trabalho necessário para atingir o objetivo da Sprint.

Sempre que um novo trabalho é necessário, o Time de Desenvolvimento adiciona este ao Backlog da Sprint. Conforme o trabalho é realizado ou completado, a estimativa do trabalho restante é atualizada. Quando elementos do plano são considerados desnecessários, eles são removidos. Somente o Time de Desenvolvimento pode alterar o Backlog da Sprint durante a Sprint. O Backlog da Sprint é altamente visível, uma imagem em tempo real do trabalho que o Time de Desenvolvimento planeja completar durante a Sprint, e que pertence exclusivamente ao Time de Desenvolvimento.

#### Monitorando o Progresso da Sprint

Em qualquer ponto do tempo na Sprint, o total do trabalho remanescente dos itens do Backlog da Sprint pode ser somado. O Time de Desenvolvimento monitora o total do trabalho restante pelo menos a cada Reunião Diária para projetar a probabilidade de alcançar o objetivo da Sprint. Ao acompanhar o trabalho restante ao longo de toda a Sprint, o Time de Desenvolvimento pode gerenciar o seu progresso.

#### [Guia Scrum - Versão 2020]



O Sprint Backlog é composto pela Meta da Sprint (por que), o conjunto de itens do Product Backlog selecionados para a Sprint (o que), bem como um plano de ação para entregar o Incremento (como). O Sprint Backlog é um plano feito por e para os Developers. É uma imagem altamente visível, em tempo real do trabalho que os Developers planejam realizar durante a Sprint para atingir a Meta da Sprint. Consequentemente, o Sprint Backlog é atualizado ao longo da Sprint conforme mais é aprendido. Deve ter detalhes suficientes para que eles possam inspecionar seu progresso na Daily Scrum.

#### **Compromisso: Meta da Sprint**

A Meta da Sprint é o único objetivo da Sprint. Embora a Meta da Sprint seja um compromisso dos Developers, esta fornece flexibilidade em termos do trabalho exato necessário para alcançá-la. A Meta da Sprint também cria coerência e foco, encorajando o Scrum Team a trabalhar junto ao invés de iniciativas separadas. A Meta da Sprint é criada durante o evento Sprint Planning e então adicionada ao Sprint Backlog. Conforme os Developers trabalham durante a Sprint, eles mantêm a Meta da Sprint em mente. Se o trabalho acabar sendo diferente do que eles esperavam, eles colaboram com o Product Owner para negociar o escopo do Sprint Backlog dentro da Sprint sem afetar a Meta da Sprint.

**(TRT-RS – 2015)** No Scrum, a lista de funcionalidades a serem implementadas em cada projeto que apresenta uma visão dos requisitos de forma mais voltada à maneira como a equipe vai desenvolvê-los, e não em uma visão de alto nível voltada às necessidades diretas do cliente, é conhecida como:

- a) product backlog.
- b) scrum backlog.
- c) sprint backlog.
- d) daily backlog.
- e) daily sprint.

**Comentários:** a questão fala que não se trata de uma lista de alto nível, logo se trata de uma lista mais detalhada. Dito isso, estamos falando da Sprint Backlog. Lembrando que o Product Backlog apresenta uma lista de alto nível (pouco detalhada) e a Sprint Backlog apresenta uma lista de baixo nível (muito detalhada) (Letra C).

**(UFRJ – 2018)** De acordo com o framework Scrum, o artefato descrito como uma lista de tarefas que o Scrum Team se compromete a fazer em um Sprint é denominado:

- a) Product Backlog.
- b) Sprint Backlog.
- c) Burndown chart.
- d) User stories.
- e) Tasks.

**Comentários:** a lista de tarefas que a equipe se compromete a fazer em uma sprint é a Sprint Backlog. A minha única ressalva é que o compromisso de fazer essa lista de tarefas é da Equipe de Desenvolvimento [Versão 2017] ou Desenvolvedores [Versão 2020] – não se trata de um compromisso do Scrum Team (Letra B).



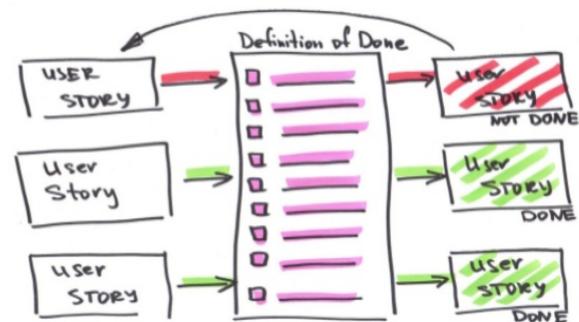
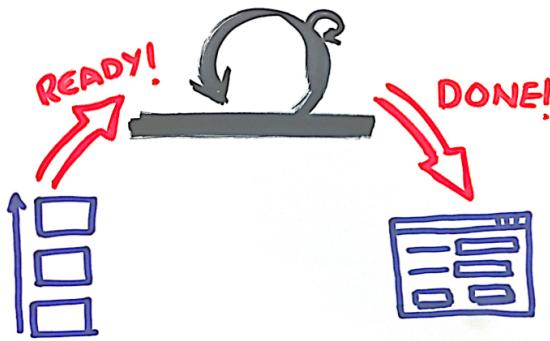
## Product Increment

### PRODUCT INCREMENT

Trata-se da soma de todos os itens do Backlog do Produto completados durante a Sprint e o valor dos incrementos de todas as sprints anteriores – sendo validado como “pronto”.

**Ao final de cada sprint, os desenvolvedores entregam um incremento do produto – resultado do que foi produzido ao final do trabalho realizado na sprint.** Esse é um dos principais conceitos do framework e vai ao encontro da sua natureza empírica, já que permite ao *Product Owner* perceber o valor do investimento realizado e também vislumbrar outras possibilidades de novos incrementos.

Para os desenvolvedores, é importante entender que o incremento deve ser algo potencialmente entregável ou liberável. *Por que potencialmente?* Porque o cliente pode optar por disponibilizar de imediato o incremento ou não. **A equipe, portanto, deve produzir código que tenha qualidade – e, então, chegamos à Definition of Done (DoD).** O que seria isso, professor? Calma, eu vou explicar...



Pronto significa pronto mesmo! Quando uma equipe ágil diz que uma funcionalidade está pronta, significa que não tem aquele “veja bem...” ou “só falta uma coisinha, mas já está pronto...”. **O DoD é um acordo formal que define claramente quais são os passos mínimos para a conclusão de um item ou funcionalidade potencialmente entregável.** Trata-se de uma lista de verificação de atividades necessárias para que um incremento seja considerado como completo.

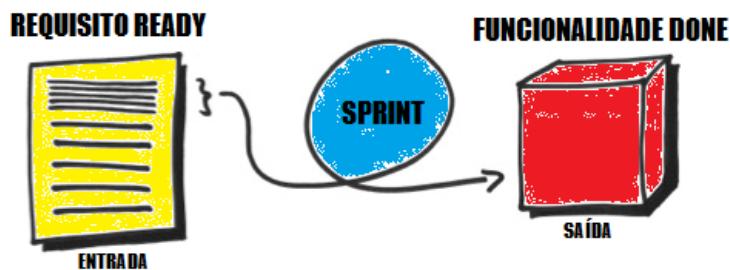
Ele serve, mais ou menos, como um contrato entre os desenvolvedores e o *Product Owner*, garantindo que todo produto gerado pelo projeto estará dentro dos padrões de qualidade estabelecidos anteriormente. Vocês devem se lembrar que a *Definition of Ready* (DoR) é um checklist de critérios acordados para que os desenvolvedores possam aceitar um requisito. *Entenderam direitinho?*

**Aqui acontece o contrário: trata-se de um checklist de critérios acordados para que o *Product Owner* possa aceitar uma funcionalidade.** Ambos tratam de critérios de aceite, mas o primeiro trata dos critérios de aceite das histórias de usuário pelos desenvolvedores e o segundo trata dos



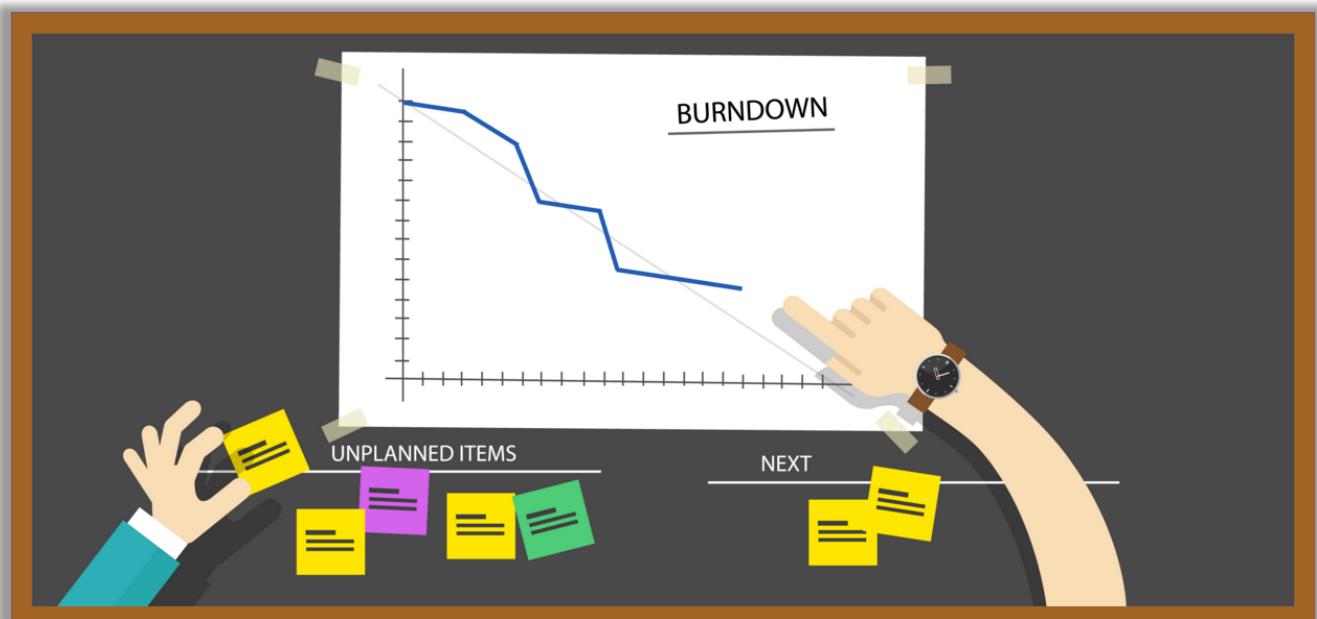
critérios de aceite das funcionalidades pelo Product Owner. Em suma: toda a Equipe Scrum deve entender o que significa “*pronto*” para ambos os casos.

**Uma funcionalidade só é considerada “*pronta*” se tiver passado por todas as etapas definidas pelos desenvolvedores (Ex: codificado, passado por todos os testes unitários, passado pelos testes de aceitação, entre outros).** Uma funcionalidade que não esteja “*pronta*” ao final da sprint deve retornar ao *Product Backlog* para que seja incluída em uma próxima sprint. Esse critério é bastante específico, cada um escolhe o seu!



Por outro lado, é uma boa prática revisar essas definições de “*pronto*” a cada sprint porque elas podem mudar ao longo do tempo. **O amadurecimento organizacional e a habilidade da equipe de resolver impedimentos podem fazer com que alguns itens sejam acrescentados com o passar do tempo.** Sempre lembrando que o *Definition of Ready* é opcional, já o *Definition of Done* é obrigatório. *Compreenderam?*

Por fim, é interessante mencionar outros artefatos que não estão explícitos no guia como o *Gráfico Burndown*, que torna visível a evolução diária do trabalho dos desenvolvedores, na medida em que mostra a comparação de produtividade entre o trabalho estimado inicialmente com a quantidade restante estimada de trabalho. **Via de regra, as unidades utilizadas são de esforço (em horas) planejado pelo tempo decorrido.**



Esse gráfico ajuda os gestores a acompanharem o andamento da equipe em projetos, considerando tempo, esforço e prazo de entrega.

#### **[Guia Scrum - Versão 2017]**

O incremento é a soma de todos os itens do Backlog do Produto completados durante a Sprint e o valor dos incrementos de todas as Sprints anteriores. Ao final da Sprint um novo incremento deve estar "Pronto", o que significa que deve estar na condição de ser utilizado e atender a definição de "Pronto" do Time Scrum. Um incremento é uma parte principal inspecionável de trabalho pronto que suporta empirismo no final da sprint. O incremento é um passo na direção de uma visão ou de um objetivo. O incremento deve estar na condição de ser utilizado independente do Product Owner decidir por liberá-lo ou não.

#### **[Guia Scrum - Versão 2020]**

Um incremento é um trampolim concreto em direção a Meta do produto. Cada incremento é adicionado a todos os incrementos anteriores e completamente verificado, garantindo que todos os incrementos funcionem juntos. A fim de fornecer valor, o incremento deve ser utilizável. Vários incrementos podem ser criados em uma Sprint. A soma dos incrementos é apresentada na Sprint Review, apoiando assim o empirismo. No entanto, um incremento pode ser entregue aos stakeholders antes do final da Sprint. A Sprint Review nunca deve ser considerada um marco para liberar valor. O trabalho não pode ser considerado parte de um incremento a menos que atenda a Definição de Pronto.

#### **Compromisso: Definição de Pronto**

A Definição de Pronto é uma descrição formal do estado do Incremento quando ela atende às medidas de qualidade exigidas para o produto. No momento em que um item do Product Backlog atende a Definição de Pronto, um incremento nasce. 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 Product Backlog não atender à Definição de Pronto, ele não poderá ser liberado ou mesmo apresentado na Sprint Review. Em vez disso, ele retorna ao Product Backlog para consideração futura.

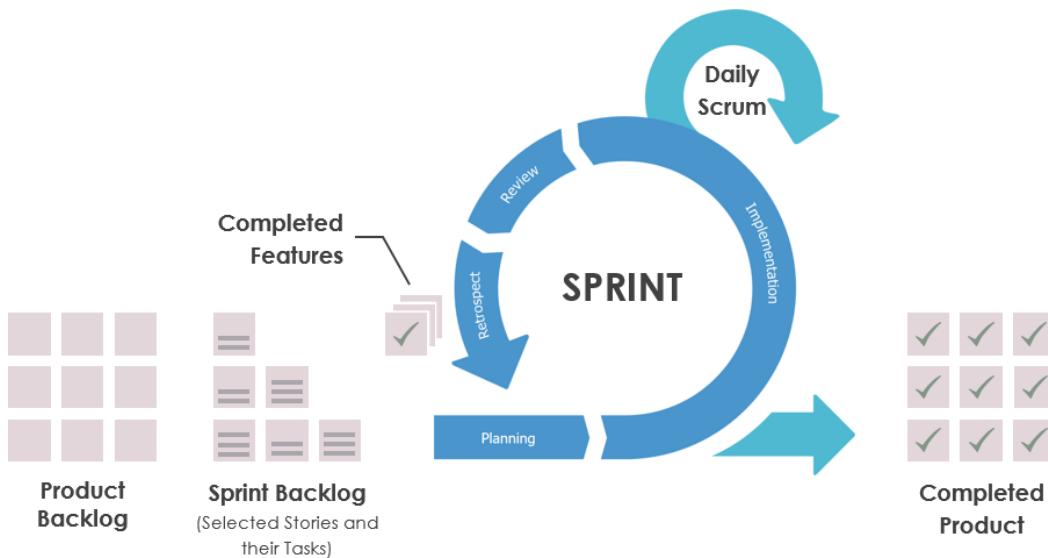
Se a Definição de Pronto para um incremento faz parte dos padrões da organização, todos os Scrum Teams devem seguir a como mínimo. Se não for um padrão organizacional, o Scrum Team deve criar uma Definição de Pronto apropriada para o produto. Os Developers devem estar em conformidade com a Definição de Pronto. Se houver vários Scrum Teams trabalhando juntos em um produto, eles devem definir e cumprir mutuamente a mesma Definição de Pronto.



## Eventos

INCIDÊNCIA EM PROVA: ALTÍSSIMA

Vamos falar agora sobre os eventos – também chamados em questões de provas de reuniões ou cerimônias! Vamos começar pelo contêiner que cobre todos os outros eventos: **Sprint**.



Pensem comigo: estou em um projeto cujo objetivo é criar uma página de *e-commerce* para uma empresa de varejo. Eu tenho que realizar diversas tarefas para construir essa página, como por exemplo criar uma funcionalidade que permita o pagamento via cartão de crédito e débito. **Essa funcionalidade pode ser (em conjunto com outras) a unidade de trabalho que satisfaz um requisito de negócio, logo pode ser realizada em uma sprint.**

**O Scrum prega que, ao fim de cada sprint, deve-se entregar um incremento potencialmente funcional do produto ao cliente.** O que seria potencialmente funcional? É aquilo que tem potencial de entrar ser utilizado pelo cliente em seu ambiente. As sprints têm duração de até um mês, permitindo feedbacks constantes quanto ao que está sendo desenvolvido. Ela é como um contêiner para todos os outros eventos e cerimônias que veremos à frente.

Eu sei que está um pouco abstrato, então vamos pensar em uma metáfora! Imagina que você contrate um marceneiro para construir os armários do apê novo que você comprou logo após passar em um concurso público. Há duas maneiras de fazer isso: se fôssemos utilizar um método tradicional, ele perguntaria como você quer os armários, passaria alguns meses construindo e algum dia montaria todos os armários na sua casa de uma só vez – **sem interações com você**.

**No método ágil, nós vamos dividir esse projeto de construção dos armários em vários ciclos de tempo fixo.** Por exemplo: nós vamos combinar com o marceneiro que a cada quinze dias eu quero que ele entregue um incremento dos armários já potencialmente funcional, ou seja, que você já possa utilizar na sua casa. Nos primeiros quinze dias, ele deve entregar os armários do banheiro prontinhos para você utilizar.



Nos próximos quinze dias, ele deve entregar os armários da área de serviço também prontos para utilizar. **Nos outros quinze, não será possível entregar todos os armários do quarto, mas ele deve entregar pelo menos o guarda-roupa já pronto para você utilizar.** Vocês conseguem notar que a cada intervalo regular de quinze dias, você vai recebendo incrementos potencialmente funcionais? Com o software é a mesma coisa...

*Qual é a grande vantagem dessa segunda opção em relação à primeira?* Bem, primeiro você não morre de ansiedade de receber os móveis somente ao final; a segunda vantagem é que você pode mudar de ideia no meio do caminho e pedir para ele mudar o projeto. *Enfim, o que importa disso tudo que eu disse? Importa que esses ciclos regulares de tempo fixo de desenvolvimento de um incremento potencialmente funcional são conhecidos também como Sprint!*

Bem, nós já sabemos que uma sprint dura um mês ou menos e se inicia imediatamente após a conclusão da sprint anterior. **Durante a sprint, é proibido realizar mudanças que coloquem em risco os objetivos da própria sprint.** Na nossa metáfora, se eu planejei que nessa sprint eu vou construir os armários do banheiro utilizando madeira do tipo cedro, eu não posso no meio do caminho alterar para madeira do tipo mogno porque isso coloca em risco os objetivos da sprint.

*Colocaria em risco, Diego?* Sim, porque essa mudança poderia inviabilizar a entrega do produto na data acordada e a sprint poderia falhar! **Além disso, é proibido mudar a composição da equipe ou diminuir as metas de qualidade.** Apesar disso, o escopo pode ser sempre clarificado, esclarecido e renegociado entre o Product Owner e os desenvolvedores durante a própria execução da sprint em andamento.

Aliás, nada impede que uma sprint seja cancelada antes de seu time-box terminar e isso somente pode ser feito pelo Product Owner (sob influência de stakeholders, desenvolvedores, etc). *Professor, por que alguém faria esse cancelamento?* **A Sprint poderá ser cancelada se o objetivo da Sprint se tornar obsoleto. Isto pode ocorrer se a organização mudar sua direção ou se as condições de mercado tecnologias mudarem.**

Geralmente, a Sprint deve ser cancelada se ela não faz mais sentido às dadas circunstâncias. No entanto, devido a curta duração da Sprint, raramente cancelamentos fazem sentido. Se uma parte do trabalho estiver potencialmente utilizável, tipicamente o Product Owner o aceita. Todos os itens de Backlog do Produto incompletos são reestimados e colocados de volta no Backlog do Produto. **O trabalho feito se deprecia rapidamente e deve ser frequentemente reestimado.**

**O cancelamento de Sprints consome recursos, já que todos tem que se reagrupar em outra reunião de planejamento da sprint para iniciar outra sprint.** Cancelamentos de sprints são frequentemente traumáticos para a Equipe Scrum, e são muito incomuns. Por falar em planejamento da sprint, vamos falar agora sobre os eventos. *Por que eu fiz essa pausa para falar um pouco mais sobre as sprints?*



Porque os quatro eventos que serão detalhados a seguir compõem uma sprint – além, é claro, do próprio trabalho de desenvolvimento. **Eventos Scrum são eventos time-boxed – o que significa que eles possuem uma duração máxima predefinida.** Eles são utilizados para criar uma rotina e minimizar a necessidade de reuniões não definidas pelo Scrum. Esses eventos são: (1) Reunião de Planejamento da Sprint; (2) Reunião Diária; (3) Revisão da Sprint; (4) Retrospectiva da Sprint.

#### [Guia Scrum - Versão 2017]

*Eventos prescritos são usados no Scrum para criar uma regularidade e minimizar a necessidade de reuniões não definidas no Scrum. Todos os eventos são eventos time-boxed, de tal modo que todo evento tem uma duração máxima. Uma vez que a Sprint começa, sua duração é fixada e não pode ser reduzida ou aumentada. Os eventos restantes podem terminar sempre que o propósito do evento é alcançado, garantindo que uma quantidade adequada de tempo seja gasta não permitindo desperdícios no processo.*

*Além da Sprint, que é um container para outros eventos, cada evento no Scrum é uma oportunidade de inspecionar e adaptar alguma coisa. Estes eventos são especificamente projetados para permitir uma transparência e inspeção criteriosa. Falhas na inclusão de qualquer um destes eventos resultará na redução da transparência e na perda de oportunidades para inspecionar e adaptar.*

*O coração do Scrum é a Sprint, um time-boxed de um mês ou menos, durante o qual um "Pronto", incremento de produto potencialmente liberável é criado. Sprints tem durações consistentes ao longo de todo o esforço de desenvolvimento. Uma nova Sprint inicia imediatamente após a conclusão da Sprint anterior. As Sprints contêm e consistem de um planejamento da Sprint, reuniões diárias, o trabalho de desenvolvimento, uma revisão da Sprint e uma retrospectiva da Sprint. Durante a Sprint:*

- Não são feitas mudanças que possam por em perigo o objetivo da Sprint;
- As metas de qualidade não diminuem; e,
- O escopo pode ser clarificado e renegociado entre o Product Owner e o Time de Desenvolvimento quanto mais for aprendido.

*Cada Sprint pode ser considerada um projeto com horizonte não maior que um mês. Como os projetos, as Sprints são utilizadas para realizar algo. Cada Sprint tem uma meta do que é para ser construído, um plano previsto e flexível que irá guiar a construção, o trabalho e o produto resultante do incremento. Sprints são limitadas a um mês corrido. Quando o horizonte da Sprint é muito longo, a definição do que será construído pode mudar, a complexidade pode aumentar e o risco pode crescer. Sprints permitem previsibilidade que garante a inspeção e adaptação do progresso em direção à meta pelo menos a cada mês corrido. Sprints também limitam o risco ao custo de um mês corrido.*

*Uma Sprint pode ser cancelada antes do time-boxed da Sprint terminar. Somente o Product Owner tem a autoridade para cancelar a Sprint, embora ele (ou ela) possa fazer isso sob influência das partes interessadas, do Time de Desenvolvimento ou do Scrum Master. A Sprint poderá ser cancelada se o objetivo da Sprint se tornar obsoleto. Isto pode ocorrer se a organização mudar sua direção ou se as condições do mercado ou das tecnologias mudarem. Geralmente a Sprint deve ser cancelada se ela não faz mais sentido às dadas circunstâncias.*

*No entanto, devido a curta duração da Sprint, raramente cancelamentos fazem sentido. Quando a Sprint é cancelada, qualquer item de Backlog do Produto completado e "Pronto" é revisado. Se uma parte do trabalho estiver potencialmente liberável, tipicamente o Product Owner o aceita. Todos os itens de Backlog do Produto incompletos são reestimados e colocados de volta no Backlog do Produto. O trabalho feito se deprecia rapidamente e deve ser frequentemente reestimado.*



O cancelamento de Sprints consome recursos, já que todos se reagrupam em outro planejamento da Sprint para iniciar outra Sprint. Cancelamentos de Sprints são frequentemente traumáticos para o Time Scrum, e são muito incomuns.

#### **[Guia Scrum - Versão 2020]**

A Sprint é um contêiner para todos os outros eventos. Cada evento no Scrum é uma oportunidade formal para inspecionar e adaptar os artefatos do Scrum. Esses eventos são projetados especificamente para permitir a transparência necessária. A falha em operar quaisquer eventos conforme prescrito resulta em oportunidades perdidas de inspeção e adaptação. Os eventos são usados no Scrum para criar regularidade e minimizar a necessidade de reuniões não definidas no Scrum. O ideal é que todos os eventos sejam realizados no mesmo horário e local para reduzir a complexidade.

Sprints são o coração do Scrum, onde ideias são transformadas em valor. São eventos de duração fixa de um mês ou menos para criar consistência. Uma nova Sprint começa imediatamente após a conclusão da Sprint anterior. Todo o trabalho necessário para atingir a meta do Produto, incluindo Sprint Planning, Daily Scrums, Sprint Review e Sprint Retrospective, acontece dentro de Sprints. Durante a Sprint:

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

Sprints permitem previsibilidade, garantindo a inspeção e adaptação do progresso em direção a uma meta do Produto ao menos uma vez por mês. Quando o horizonte de uma Sprint é muito longo, a meta da Sprint pode se tornar inválida, a complexidade pode aumentar e o risco pode aumentar. Sprints mais curtas podem ser empregados para gerar mais ciclos de aprendizagem e limitar os riscos de custo e esforço a um período de tempo menor. Cada Sprint pode ser considerado um projeto curto.

Existem várias práticas para prever o progresso, como burn-downs, burn-ups ou cumulative flows. Embora comprovadamente úteis, eles não substituem a importância do empirismo. Em ambientes complexos, o que acontecerá é desconhecido. Somente o que já aconteceu pode ser usado para a tomada de decisão voltada para o futuro. Uma Sprint pode ser cancelada se a Meta da Sprint se tornar obsoleta. Apenas o Product Owner tem autoridade para cancelar a Sprint.



## Sprint Planning

O trabalho a ser realizado na Sprint é planejado na Reunião de Planejamento da Sprint. Este planejamento é criado com o trabalho colaborativo de toda a Equipe Scrum. Ela possui um *time-box* com no máximo oito horas para uma sprint de um mês de duração – para sprints menores, a duração é menor. O Scrum Master garante que o evento ocorra e que os participantes entendam seu propósito. **Ela consiste em duas partes e devem responder adequadamente as perguntas:**

**1. O QUE SERÁ ENTREGUE COMO RESULTADO DO INCREMENTO DA PRÓXIMA SPRINT?**

**2. COMO O TRABALHO NECESSÁRIO PARA ENTREGAR O INCREMENTO SERÁ REALIZADO?**

**Na primeira parte, os desenvolvedores tentam prever as funcionalidades que serão desenvolvidas durante a sprint.** O *Product Owner* apresenta as histórias de usuário mais priorizados do *Product Backlog* aos desenvolvedores. *Como ele faz isso?* Em geral, ele dá um valor de negócio para cada item do backlog, organizando-os em forma decrescente de valor de negócio. *Como assim, professor?*

Imagine que exista um item que o *Product Owner* deseja muito que seja implementado – ele pode dizer que esse item tem o valor de negócio de 1000. Agora imagine que exista outro item no *Product Backlog* que o *Product Owner* não liga tanto – ele dá um valor de negócio de 10. **Dessa forma, o *Product Owner* consegue ordenar os itens de acordo com o valor de negócio.** Feito isso, é hora de estimar o esforço de desenvolvimento de cada item do backlog.

Quando nós utilizamos histórias de usuário, é comum utilizarmos uma outra unidade de medida para medir esforço, em vez do tempo – utilizado frequentemente em metodologias tradicionais. No caso de Histórias de Usuário (User Stories), nós utilizamos Pontos de História (Story Points). **Trata-se de uma unidade de medida relativa que leva em consideração o esforço<sup>1</sup> necessário para realizar uma determinada funcionalidade.**

Se uma funcionalidade requerer o dobro de esforço para ser implementada, ela receberá aproximadamente o dobro de Story Points. Para fazer essa estimativa, os desenvolvedores realizam uma comparação com outras histórias já estimadas. Caso não haja ainda nada estimado no *Product Backlog*, a equipe localiza a história de usuário com o menor esforço para desenvolvimento e o utiliza como base de comparação.

<sup>1</sup> Há uma polêmica danada: alguns afirmam que ela estima complexidade e, não, esforço; outros dizem que é uma combinação de complexidade, esforço, risco, etc.



**Uma das melhores formas de estimar Story Points é por meio de uma técnica chamada Planning Poker, que não está no guia oficial, mas que é frequentemente utilizada tanto para estimar esforço como para estimar tamanho.** Essa técnica busca uma abordagem dinâmica e divertida na estimativa de Histórias de Usuário, e elimina a influência que um desenvolvedor possa exercer sobre outros.

Feita a estimativa de cada história de usuário, os desenvolvedores decidirão a quantidade de itens do backlog a serem realizados na sprint. Galera, vamos detalhar isso um pouco mais! **Story Points é uma unidade de medida relativa que leva em consideração o esforço necessário para realizar uma determinada funcionalidade.** Se uma funcionalidade requerer o dobro de esforço para ser implementada, ela receberá aproximadamente o dobro de Story Points.

Para estimar a quantidade de Story Points de uma User Story, os desenvolvedores os comparam com outros já estimados. Caso não haja ainda nada estimado no Product Backlog, os desenvolvedores localizam o User Story com o menor esforço para o desenvolvimento, e o utiliza como base para comparações futuras. **Uma das melhores formas de se estimar Story Points é utilizando o Planning Poker.**

Isso porque se trata de uma abordagem divertida na estimativa de itens ou User Stories que elimina a influência que um membro dos desenvolvedores possa exercer sobre os outros. *Vamos entender um pouco melhor como ele funciona?* No início do Planning Poker, cada membro do time recebe um conjunto de cartas. Cada carta exibe um dos valores válidos para a estimativa (Ex: 0, 1, 2, 3, 5, 8, 13, 20, 40, e 100).

**Em geral, os valores seguem uma escala baseada na Sequência de Fibonacci.** Observem: outra sequência pode ser escolhida, porém Fibonacci é a mais utilizada. Para cada User Story a ser estimativa, o Product Owner lê a descrição e esclarece quaisquer dúvidas que os membros do time tenham. Entretanto, é importante lembrar que em determinado ponto, qualquer discussão adicional não busca uma precisão maior.

Após todas as questões serem respondidas, cada membro seleciona uma carta representando sua estimativa, mas sem mostrar aos outros. As cartas não são mostradas até o momento em que todos, simultaneamente, exibem seus valores, de forma que todos vejam os valores selecionados simultaneamente. Neste ponto, é normal que as estimativas sejam significativamente diferentes. E na realidade é um bom sinal.

**Quando as estimativas diferem, os membros expõem os motivos que os levaram a escolher aqueles valores.** Depois das explanações e discussões, todos recolhem suas cartas e estimam novamente da mesma forma. **O Planning Poker funciona porque se baseia em diversas opiniões durante a estimativa.** Como os desenvolvedores formam um time multidisciplinar, eles são os mais indicados para estimar as histórias do que qualquer outra pessoa.

**Além disso, os diálogos e justificativas permitem uma maior acuracidade das estimativas, especialmente nos itens com maior incerteza.** E isto é de extrema importância em um projeto que



tenha um certo nível de complexidade. Finalmente, estudos mostram que a média de estimativas individuais levam a melhores resultados, uma vez que promovem discussões em grupo.

Estas discussões em grupo são a base do Planning Poker e elas conduzem a um consenso entre os indivíduos participantes. **Mais uma vez: Fibonacci é a escala mais utilizada na estimativa de User Story por Planning Poker.** Isso se deve ao fato de a Sequência de Fibonacci ser uma função quadrática, em vez de uma função linear. Detalhe: algumas vezes as histórias de usuário são muito grandes para serem desenvolvidas em uma sprint – são chamadas de Épicos.

Elas precisarão ser quebradas em partes menores. Mais que isso, em alguns projetos é necessário um nível ainda maior que um épico – chamado de Saga – para features geralmente mais complexas. *Foram selecionados os itens?* Ok! Agora a Equipe Scrum definirá uma meta para a sprint que será o guia para os desenvolvedores sobre o que estará sendo desenvolvido durante a Sprint.

**Na segunda parte, os desenvolvedores decidem como irá transformar os itens selecionados em um incremento durante a Sprint e desenvolve o Sprint Backlog.** *Por falar nisso... qual a diferença entre Product e Sprint Backlog mesmo?* O primeiro é uma lista de todos os requisitos/funcionalidades de usuário levantados até o momento e mantidas pelo Product Owner, que pode alterá-las a qualquer momento.

**O segundo é um subconjunto do primeiro transformado em uma lista de tarefas técnicas e mantidas pelos desenvolvedores, que pode alterá-las a qualquer momento.** Bem, ao final do planejamento da Sprint, os desenvolvedores devem ser capazes de explicar ao Product Owner e ao Scrum Master como pretende trabalhar como equipe auto-organizada para completar o objetivo da Sprint e criar o incremento previsto.

Com o Sprint Backlog criado, define-se a meta da sprint! *O que seria isso, Diego?* Galera, a meta nada mais é que um objetivo definido para ser satisfeita ao final da sprint. O Product Owner pode estar presente durante a segunda parte da reunião para clarificar itens do Backlog do Produto. *Vocês se lembram que essa é uma das principais responsabilidades desse papel, certo?* Pois é... vamos para o nosso próximo evento...

#### [Guia Scrum - Versão 2017]

*O trabalho a ser realizado na Sprint é planejado durante o planejamento da Sprint. Este plano é criado com o trabalho colaborativo de todo o Time Scrum. O Planejamento da Sprint é um time-boxed com no máximo oito horas para uma Sprint de um mês de duração. Para Sprints menores, este evento é usualmente menor. O Scrum Master garante que o evento ocorra e que os participantes entendam seu propósito. O Scrum Master ensina o Time Scrum a manter-se dentro dos limites do time-box. O planejamento da Sprint responde as seguintes questões:*

- *O que pode ser entregue como resultado do incremento da próxima Sprint?*
- *Como o trabalho necessário para entregar o incremento será realizado?*

*Tópico Um: O que pode ser Pronto nesta Sprint?*



O Time de Desenvolvimento trabalha para prever as funcionalidades que serão desenvolvidas durante a Sprint. O Product Owner debate o objetivo que a Sprint deve realizar e os itens de Backlog do Produto que, se completados na Sprint, atingirão o objetivo da Sprint. Todo o Time Scrum colabora com o entendimento do trabalho da Sprint. A entrada desta reunião é o Backlog do Produto, o mais recente incremento do produto, a capacidade projetada do Time de Desenvolvimento durante a Sprint e o desempenho passado do Time de Desenvolvimento.

O número de itens selecionados do Backlog do Produto para a Sprint é o único trabalho do Time de Desenvolvimento. Somente o Time de Desenvolvimento pode avaliar o que pode ser completado ao longo da próxima Sprint. Durante o Planejamento da Sprint, o Time Scrum também determina a meta da Sprint. A meta da Sprint é o objetivo que será satisfeito dentro da Sprint através da implementação do Backlog do Produto, e que fornece a orientação para o Time de Desenvolvimento sobre o porquê dele estar construindo o incremento.

### **Tópico Dois: Como o trabalho escolhido será Pronto?**

Tendo definido o objetivo da Sprint e selecionado os itens de Backlog do Produto da Sprint, o Time de Desenvolvimento decide como irá construir essas funcionalidades durante a Sprint e transformá-las em um incremento de produto "Pronto". Os itens de Backlog do Produto selecionados para a Sprint, junto com o plano de entrega destes itens é chamado de Backlog da Sprint. O Time de Desenvolvimento frequentemente inicia o desenho do sistema e do trabalho necessário para converter o Backlog do Produto em um incremento funcional do produto.

O trabalho pode ser de vários tamanhos ou esforços. Contudo, o trabalho suficiente é planejado durante o planejamento da Sprint pelo Time de Desenvolvimento para prever o que este acredita que poderá fazer durante a próxima Sprint. O trabalho planejado pelo Time de Desenvolvimento para os primeiros dias da Sprint é decomposto até o final desta reunião, frequentemente em unidades de um dia de duração ou menos. O Time de Desenvolvimento se auto-organiza para realizar todo o trabalho do Backlog da Sprint, tanto durante o planejamento da Sprint quanto no que for necessário durante a Sprint.

O Product Owner pode ajudar a clarificar os itens de Backlog do Produto selecionados e nas decisões conflituosas de troca. Se o Time de Desenvolvimento determina que tem excesso ou falta de trabalho, os itens do Backlog da Sprint podem ser renegociados com o Product Owner. O Time de Desenvolvimento também pode convidar outras pessoas para participar desta reunião para fornecer opinião técnica ou de domínios específicos. No final do planejamento da Sprint, o Time de Desenvolvimento deve ser capaz de explicar ao Product Owner e ao Scrum Master como pretende trabalhar como equipe auto-organizada para completar o objetivo da Sprint e criar o incremento previsto.

### **Meta da Sprint**

A meta da Sprint é um objetivo definido para a Sprint que pode ser satisfeito através da implementação do Backlog do Produto. Este fornece uma direção para o Time de Desenvolvimento sobre o porquê de estar construindo o incremento. Este é criado durante a reunião de planejamento da Sprint. O objetivo da Sprint dá ao Time de Desenvolvimento alguma flexibilidade a respeito da funcionalidade que será completada dentro da Sprint. Os itens do Backlog do Produto selecionados entregam uma função coerente, que pode ser o objetivo da Sprint.

O objetivo da Sprint pode ser qualquer outro coerente que faça o Time de Desenvolvimento trabalhar em conjunto em vez de em iniciativas separadas. Conforme o Time de Desenvolvimento trabalha, eles mantêm o objetivo da Sprint em mente. A fim de satisfazer o objetivo da Sprint, implementando funcionalidade e tecnologia. Caso o trabalho acabe por ser diferente do esperado pelo Time de Desenvolvimento, então eles colaboram com o Product Owner para negociar o escopo do Backlog da Sprint dentro da Sprint.

**[Guia Scrum - Versão 2020]**



A Sprint Planning inicia a Sprint ao definir o trabalho a ser realizado na Sprint. Este plano resultante é criado pelo trabalho colaborativo de todo o Scrum Team. O Product Owner garante que os participantes estejam preparados para discutir os itens mais importantes do Product Backlog e como eles são mapeados para a Meta do Produto. O Scrum Team também pode convidar outras pessoas para participar da Sprint Planning para fornecer conselhos.

**A Sprint Planning aborda os seguintes tópicos:**

**Tópico um: Por que esta Sprint é valiosa?**

O Product Owner propõe como o produto pode aumentar seu valor e utilidade na Sprint atual. Todo o Scrum Team então colabora para definir uma Meta da Sprint que comunica porque a Sprint é valiosa para os stakeholders. A meta da Sprint deve ser finalizada antes do final da Sprint Planning.

**Tópico dois: O que pode ser feito nesta Sprint?**

Por meio de discussão com o Product Owner, os Developers selecionam itens do Product Backlog para incluir na Sprint atual. O Scrum Team 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 Developers sabem sobre seu desempenho anterior, sua capacidade futura e sua Definição de Pronto, mais confiantes eles estarão em suas previsões quanto a Sprint.

**Tópico três: Como o trabalho escolhido será realizado?**

Para cada item do Product Backlog selecionado, os Developers planejam o trabalho necessário para criar um Incremento que atenda à Definição de Pronto. Isso geralmente é feito decompondo itens do Product Backlog em itens de trabalho menores de um dia ou menos. A forma como isso é feito fica a critério exclusivo dos Developers. Ninguém mais diz a eles como transformar itens do Product Backlog em incrementos de valor. A Meta da Sprint, os itens do Product Backlog selecionados para a Sprint, mais o plano para entregá-los são chamados juntos de Sprint Backlog.

A Sprint Planning tem um Timebox definido com duração máxima de oito horas para uma Sprint de um mês. Para Sprints mais curtas, o evento geralmente é mais curto.

**(TRE-BA – 2017)** A reunião de planejamento da sprint do Scrum é o evento em que:

- a) é definida a equipe scrum e são cancelados os itens da sprint anterior que não tenham sido entregues e os que tenham sido entregues, mas tenham sido rejeitados pelo usuário.
- b) são escritas as histórias dos usuários por meio do planning poker.
- c) é definida a meta da sprint e são selecionados os itens do product backlog que comporão a sprint.
- d) é decidido pelo PO (product owner) se haverá o cancelamento ou não da sprint em curso.
- e) participam exclusivamente o PO (product owner) e o SM (scrum master), que, por meio do planning poker, priorizaram os itens do backlog.



**Comentários:** (a) Errado, a equipe é definida antes e não se cancela nada; (b) Errado, *planning poker* é uma técnica de estimativa de esforço; (c) Correto; (d) Errado, se ainda está no planejamento, a sprint não está em curso; (e) Errado, os desenvolvedores também participam da reunião de planejamento (Letra C).



## Daily Scrum

**A Reunião Diária (15 minutos) é um evento que busca criar um plano para as próximas 24 horas e inspecionar o trabalho desde a última Reunião Diária.** É realizada em todos os dias da sprint e é o evento em que os desenvolvedores planejam o trabalho para as próximas 24 horas – isso otimiza a colaboração e a performance do time através da inspeção do trabalho desde a última Reunião Diária, e da previsão do próximo trabalho da Sprint.

A Reunião Diária é mantida no mesmo horário e local todo dia para reduzir a complexidade e não deve ocorrer necessariamente em pé. **A Reunião Diária aumenta a probabilidade dos desenvolvedores atingirem o objetivo da Sprint.** Todos os dias, os desenvolvedores devem entender como o mesmo pretende trabalhar em conjunto, como um time auto-organizado, para completar o objetivo da Sprint e criar o incremento previsto até o final da Sprint.

**A estrutura da reunião é definida pelos desenvolvedores e pode ser conduzida de diferentes formas desde que estas se foquem no progresso em direção à meta da sprint.** Alguns Times de Desenvolvimento utilizarão perguntas, outros se basearão em discussões – no meu caso pessoal, nós utilizávamos ambos: perguntas e discussões. Bem, aqui segue um exemplo do que pode ser utilizado:

**1. O QUE EU FIZ ONTEM QUE AJUDOU OS DESENVOLVEDORES A ATENDEREM A META DA SPRINT?**

**2. O QUE EU FAREI HOJE PARA AJUDAR OS DESENVOLVEDORES ATENDEREM A META DA SPRINT?**

**3. EU VEJO ALGUM OBSTÁCULO QUE IMPEÇA A MIM OU OS DESENVOLVEDORES NO ATENDIMENTO DA META DA SPRINT?**

**As Reuniões Diárias melhoram a comunicação entre os integrantes,** eliminam a necessidade de outras reuniões, identificam e removem impedimentos, destacam e promovem rápidas tomadas de decisão, e melhoram o nível de conhecimento da Equipe. Apesar de poder contar com a presença de outras partes interessadas, essa é uma reunião feita pelos desenvolvedores para os próprios desenvolvedores.

**[Guia Scrum - Versão 2017]**

A Reunião Diária do Scrum é um evento time-boxed de 15 minutos para o Time de Desenvolvimento. A Reunião Diária é realizada em todos os dias da Sprint. Nela o Time de Desenvolvimento planeja o trabalho para as próximas 24 horas. Isso optimiza a colaboração e a performance do time através da inspeção do trabalho desde a última Reunião Diária, e da previsão do próximo trabalho da Sprint. A Reunião Diária é mantida no mesmo horário e local todo dia para reduzir a complexidade.



O Time de Desenvolvimento usa a Reunião Diária para inspecionar o progresso em direção ao objetivo da Sprint e para inspecionar se o progresso tende na direção de completar o trabalho do Backlog da Sprint. A Reunião Diária aumenta a probabilidade do Time de Desenvolvimento atingir o objetivo da Sprint. Todos os dias, o Time de Desenvolvimento deve entender como o mesmo pretende trabalhar em conjunto, como um time auto-organizado, para completar o objetivo da Sprint e criar o incremento previsto até o final da Sprint.

A estrutura da reunião é definida pelo Time de Desenvolvimento e pode ser conduzida de diferentes formas desde que estas foquem no progresso em direção à Meta da Sprint. Alguns Times de Desenvolvimento utilizarão perguntas, outros se basearão em discussões. Aqui segue um exemplo do que pode ser utilizado:

- O que eu fiz ontem que ajudou o Time de Desenvolvimento a atingir a meta da Sprint?
- O que eu farei hoje para ajudar o Time de Desenvolvimento atingir a meta da Sprint?
- Eu vejo algum obstáculo que impeça a mim ou o Time de Desenvolvimento no atingimento da meta da Sprint?

O Time de Desenvolvimento ou membros da equipe frequentemente se encontram imediatamente após a Reunião Diária para discussões detalhadas, ou para adaptar, ou replanejar, o restante do trabalho da Sprint. O Scrum Master assegura que o Time de Desenvolvimento tenha a reunião, mas o Time de Desenvolvimento é responsável por conduzir a Reunião Diária. O Scrum Master ensina o Time de Desenvolvimento a manter a Reunião Diária dentro do time-box de 15 minutos.

A Reunião Diária é uma reunião interna do Time de Desenvolvimento. Se outros estiverem presentes, o Scrum Master deve garantir que eles não perturbem a reunião. Reuniões Diárias melhoram as comunicações, eliminam outras reuniões, identificam e removem impedimentos para o desenvolvimento, destacam e promovem rápidas tomadas de decisão, e melhoram o nível de conhecimento do Time de Desenvolvimento. Esta é uma reunião chave para inspeção e adaptação.

#### **[Guia Scrum - Versão 2020]**

O propósito da Daily Scrum é inspecionar o progresso em direção a Meta da Sprint e adaptar o Sprint Backlog conforme necessário, ajustando o próximo trabalho planejado. A Daily Scrum é um evento de 15 minutos para os Developers do Scrum Team. Para reduzir a complexidade, é realizado no mesmo horário e local, todos os dias úteis da Sprint. Se o Product Owner ou o Scrum Master estão trabalhando ativamente nos itens do Sprint Backlog, eles participam como Developers.

Os Developers podem selecionar qualquer estrutura e técnicas que quiserem, desde que seu Daily Scrum se concentre no progresso em direção a Meta da Sprint e produza um plano de ação para o próximo dia de trabalho. Isso cria foco e melhora o autogerenciamento. As Daily Scrums melhoram as comunicações, identificam os impedimentos, promovem a rápida tomada de decisões e consequentemente, eliminam a necessidade de outras reuniões. A Daily Scrum não é o único momento em que os Developers 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.

**(EBSERH – 2017)** As reuniões diárias estabelecidas pelos autores do SCRUM (Jeff Sutherland e Ken Schwaber) faz com que o Scrum Master oriente o Time de Desenvolvimento a manter a Reunião Diária dentro do time-box constante de:

- a) 15 minutos      b) 1 hora      c) 30 minutos      d) 2 horas      e) 45 minutos

**Comentários:** o time-box da reunião diária é de 15 minutos (Letra A).



## Sprint Review

**No final da sprint, ocorre a Revisão da Sprint (Proporcional a 4 horas).** Embora seja utilizada para demonstrar as novas funcionalidades desenvolvidas durante a sprint, seu principal motivo é o de inspecionar o que os desenvolvedores produziram e colher opiniões e impressões dos presentes para, caso seja necessário, adaptar o plano para a sprint seguinte. O foco aqui é aprimorar o produto!



Vocês se lembram do filme *O Gladiador*? Pois é! A Revisão da Sprint é o momento em que o Product Owner valida (👍) ou não (👎) a sprint, de acordo com a meta que tenha sido acordada com os desenvolvedores durante a reunião de planejamento da sprint. **Discute-se os problemas e as soluções e, após a demonstração do incremento, respondem-se quaisquer dúvidas dos presentes.**

### [Guia Scrum - Versão 2017]

A Revisão da Sprint é realizada no final da Sprint para inspecionar o incremento e adaptar o Backlog do Produto se necessário. Durante a Revisão da Sprint o Time Scrum e as partes interessadas colaboram sobre o que foi feito na Sprint. Com base nisso e em qualquer mudança no Backlog do Produto durante a Sprint, os participantes colaboram nas próximas coisas que podem ser feitas para otimizar valor. Esta é uma reunião informal, não uma reunião de status, e a apresentação do incremento destina-se a motivar e obter feedback e promover a colaboração.

Esta é uma reunião de no máximo 4 horas de duração para uma Sprint de um mês. Para Sprints menores, este evento é usualmente menor. O Scrum Master garante que o evento ocorra e que os participantes entendam o seu propósito. O Scrum Master ensina todos os envolvidos a manter a reunião dentro do Time-box. A Revisão da Sprint inclui os seguintes elementos:

- Os participantes incluem o Time Scrum e os Stakeholders chaves convidados pelo Product Owner;
- O Product Owner esclarece quais itens do Backlog do Produto foram “Prontos” e quais não foram “Prontos”;
- O Time de Desenvolvimento discute o que foi bem durante a Sprint, quais problemas ocorreram dentro da Sprint, e como estes problemas foram resolvidos;
- O Time de Desenvolvimento demonstra o trabalho que está “Pronto” e responde as questões sobre o incremento;



- O Product Owner discute o Backlog do Produto tal como está. Ele (ou ela) projeta os prováveis alvos e datas de entrega baseado no progresso até a data (se necessário);
- O grupo todo colabora sobre o que fazer a seguir, e é assim que a Revisão da Sprint fornece valiosas entradas para o Planejamento da Sprint subsequente;
- Revisão de como o mercado ou o uso potencial do produto pode ter mudado e o que é a coisa mais importante a se fazer a seguir; e,
- Revisão da linha do tempo, orçamento, potenciais capacidades, e mercado para a próxima versão esperada de funcionalidade ou de capacidade do produto.

O resultado da Revisão da Sprint é um Backlog do Produto revisado que define os prováveis Itens de Backlog do Produto para a próxima Sprint. O Backlog do Produto pode também ser ajustado completamente para atender novas oportunidades.

#### **[Guia Scrum - Versão 2020]**

O propósito da Sprint Review é inspecionar o resultado da Sprint e determinar as adaptações futuras. O Scrum Team apresenta os resultados de seu trabalho para os principais stakeholders e o progresso em direção a Meta do Produto é discutido. Durante o evento, o Scrum Team e os stakeholders revisam o que foi realizado na Sprint e o que mudou em seu ambiente. Com base nessas informações, os participantes colaboram sobre o que fazer a seguir. O Product Backlog também pode ser ajustado para atender a novas oportunidades.

A Sprint Review é uma sessão de trabalho e o Scrum Team deve evitar limitá-la a uma apresentação. A Sprint Review é o penúltimo evento da Sprint e tem um Timebox com prazo máximo de quatro horas para uma Sprint de um mês. Para Sprints mais curtas, o evento geralmente é mais curto.

**(TRE-SP – 2017)** Considere, por hipótese, que uma equipe de Analistas do TRE-SP participou de uma reunião de um projeto baseado no Scrum e, ao final, o Backlog do Produto foi revisto e completamente ajustado para atender às novas necessidades de verificação de contribuições para campanhas de candidatos, advindas de pessoas físicas sob suspeita de corrupção. Os Analistas participaram da reunião:

- a) de Revisão da Sprint.
- b) de Retrospectiva da Sprint.
- c) diária.
- d) de Verificação da Sprint.
- e) de Planejamento da Sprint.

**Comentários:** a questão trata da revisão da sprint, que é executada no final para inspecionar o incremento e adaptar o Backlog do Produto, se necessário (Letra A).



## Sprint Retrospective

A Retrospectiva da Sprint (Proporcional a 3 horas) é uma chance para o Scrum Team **inspecionar a si próprio e criar um plano de melhorias para a próxima sprint**. Ela inspeciona como foi a última sprint em relação às pessoas, às relações, aos processos e às ferramentas. Pode identificar e ordenar os itens que se tornaram potenciais de melhorias e cria um plano para implementar melhorias no trabalho.

**O Scrum Master garante que o evento seja positivo e produtivo. O Scrum Master ensina todos a manter o evento dentro do time-box.** O Scrum Master participa da reunião como um membro auxiliar do time devido a sua responsabilidade pelo processo Scrum. O Scrum Master encoraja o Time Scrum a melhorar, dentro do processo do framework do Scrum, seu processo de desenvolvimento e suas práticas para torná-lo mais efetivo e agradável para a próxima Sprint.

Durante cada Retrospectiva da Sprint, o Time Scrum planeja formas de aumentar a qualidade do produto melhorando o processo de trabalho ou adaptando a definição de “Pronto”, se apropriado e sem entrar em conflito com os padrões do produto ou organização. **Ao final da Retrospectiva da Sprint, o Time Scrum deverá ter identificado melhorias que serão implementadas na próxima Sprint.**

A implementação destas melhorias na próxima Sprint é a forma de adaptação à inspeção que o Time Scrum faz a si próprio. Apesar de que melhorias podem ser implementadas a qualquer momento, a Retrospectiva da Sprint fornece uma oportunidade formal focada em inspeção e adaptação. **Fechou? Antes disso, vamos falar de um evento não-oficial, mas que geralmente é realizado: Reunião de Visão! O que é isso, professor?**

Trata-se do momento que visa estabelecer um ponto no processo em que o Product Owner deve expor os detalhes do produto a ser construído. A saída dessa reunião deve ser uma visão sobre o produto, isto é, **representa como os clientes, usuários finais, gerentes, stakeholders, executivos, entre outros, visualizam o resultado final do produto que será criado**. Para tal, pode-se utilizar diversas técnicas como: *Product Vision Box*, *Product RoadMap* ou *Elevator Pitch Sentence*.

Vamos ver um pouco dessa última técnica! Geoffrey Moore, no seu livro *Crossing the Chasm*, apresenta um modelo interessante para a Visão do Produto, o chamado “Teste do Elevador”. **A ideia é que seja possível explicar o que é o produto durante a subida de um elevador, ou seja, em um tempo bastante curto.** Adaptado por Jim Highsmith, esse modelo tem o formato apresentado na imagem da página seguinte.

**Um exemplo de visão sobre um produto de turismo poderia ser:** “Para turistas usuários de smartphone que desejam aproveitar melhor seus locais de destino, o MyTrip é um aplicativo móvel de viagens que sugere roteiros diários flexíveis de acordo com seu perfil de viajante. Ao contrário de guias de viagens com roteiros predefinidos e burocráticos, nosso produto elabora trajetos personalizados e adaptáveis”.



## Elevator Pitch sentence structure:

FOR (target customer), WHO HAS  
(customer need), (product name) IS A  
(market category) THAT (one key benefit).  
UNLIKE (competition), THE  
PRODUCT (unique differentiator).

**Lembremos que a visão do produto, de forma geral, deve permanecer estável durante todo o projeto.** Ela é criada, gerenciada e compartilhada pelo Product Owner, que garante que o Product Backlog esteja sempre alinhado a ela. No entanto, as partes interessadas relevantes podem estar diretamente envolvidas no refinamento dessa visão. Há outra cerimônia não-oficial (apesar de muito comum) chamada Release Planning Meeting. *O que seria isso?*

Nós vimos que ao final da sprint, a equipe entrega um incremento do produto potencialmente funcional, isto é, tem o potencial de entrar em produção. Ora, muitas vezes é desejável esperar algumas sprints até juntas todas as funcionalidades e entregar uma release (conjunto de funcionalidades). Essa cerimônia serve para planejar como será essa release. Isso é muito importante, porque vocês devem saber a criticidade de colocar algo em produção.

**É comum ter várias restrições, preocupações e dependências, como datas importantes, itens contratuais, logística, entre outros.** Dessa forma, a equipe precisa planejar suas entregar várias sprints à frente. Por fim, é salutar enfatizar que o ciclo de vida do nosso framework é baseado em três fases principais:

### 1. PRÉ-PLANEJAMENTO (PRE-GAME PHASE)

Define o sistema sendo desenvolvido. Cria-se o Product Backlog, que contém os requisitos atuais e informações sobre o planejamento do projeto. Cria-se também uma arquitetura de alto nível.

### 2. DESENVOLVIMENTO (GAME PHASE)



O sistema é desenvolvido em sprints, por meio de uma abordagem iterativa. A cada sprint, novas funcionalidades são adicionadas de modo tradicional, i.e., análise, projeto, implementação, etc.

### **3. PÓS-PLANEJAMENTO (POST-GAME PHASE)**

Após o desenvolvimento, são feitas reuniões para analisar o progresso do projeto e demonstrar o software para os clientes. Aqui ocorrem as etapas de integração, testes finais e documentação.

#### **[Guia Scrum - Versão 2017]**

*A Retrospectiva da Sprint é uma oportunidade para o Time Scrum inspecionar a si próprio e criar um plano para melhorias a serem aplicadas na próxima Sprint. A Retrospectiva da Sprint ocorre depois da Revisão da Sprint e antes do planejamento da próxima Sprint. Esta é uma reunião de no máximo três horas para uma Sprint de um mês. Para Sprint menores, este evento é usualmente menor. O Scrum Master garante que o evento ocorra e que os participantes entendam seu propósito.*

*O Scrum Master garante que o evento seja positivo e produtivo. O Scrum Master ensina todos a manter o evento dentro do time-box. O Scrum Master participa da reunião como um membro auxiliar do time devido a sua responsabilidade pelo processo Scrum. O propósito da Retrospectiva da Sprint é:*

- *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; e,*
- *Criar um plano para implementar melhorias no modo que o Time Scrum faz seu trabalho;*

*O Scrum Master encoraja o Time Scrum a melhorar, dentro do processo do framework do Scrum, seu processo de desenvolvimento e suas práticas para torná-lo mais efetivo e agradável para a próxima Sprint. Durante cada Retrospectiva da Sprint, o Time Scrum planeja formas de aumentar a qualidade do produto melhorando o processo de trabalho ou adaptando a definição de "Pronto", se apropriado e sem entrar em conflito com os padrões do produto ou organização.*

*Ao final da Retrospectiva da Sprint, o Time Scrum deverá ter identificado melhorias que serão implementadas na próxima Sprint. A implementação destas melhorias na próxima Sprint é a forma de adaptação à inspeção que o Time Scrum faz a si próprio. Apesar de que melhorias podem ser implementadas a qualquer momento, a Retrospectiva da Sprint fornece uma oportunidade formal focada em inspeção e adaptação.*

#### **[Guia Scrum - Versão 2020]**

*O propósito da Sprint Retrospective é planejar maneiras de aumentar a qualidade e a eficácia. O Scrum Team inspeciona como foi a última Sprint em relação a indivíduos, interações, processos, ferramentas e sua Definição de Pronto. Os elementos inspecionados geralmente variam com o domínio de trabalho. As suposições que os desviaram são identificadas e suas origens exploradas. O Scrum Team discute o que deu certo durante a Sprint, quais problemas encontraram e como esses problemas foram (ou não) resolvidos.*

*O Scrum Team identifica as mudanças mais úteis para melhorar sua eficácia. As melhorias mais impactantes são endereçadas o mais rápido possível. Essas podem até ser adicionadas ao Sprint Backlog para a próxima Sprint. A Sprint Retrospective conclui a Sprint. É limitada pelo Timebox de no máximo três horas para uma Sprint de um mês. Para Sprints mais curtas, o evento geralmente é mais curto.*



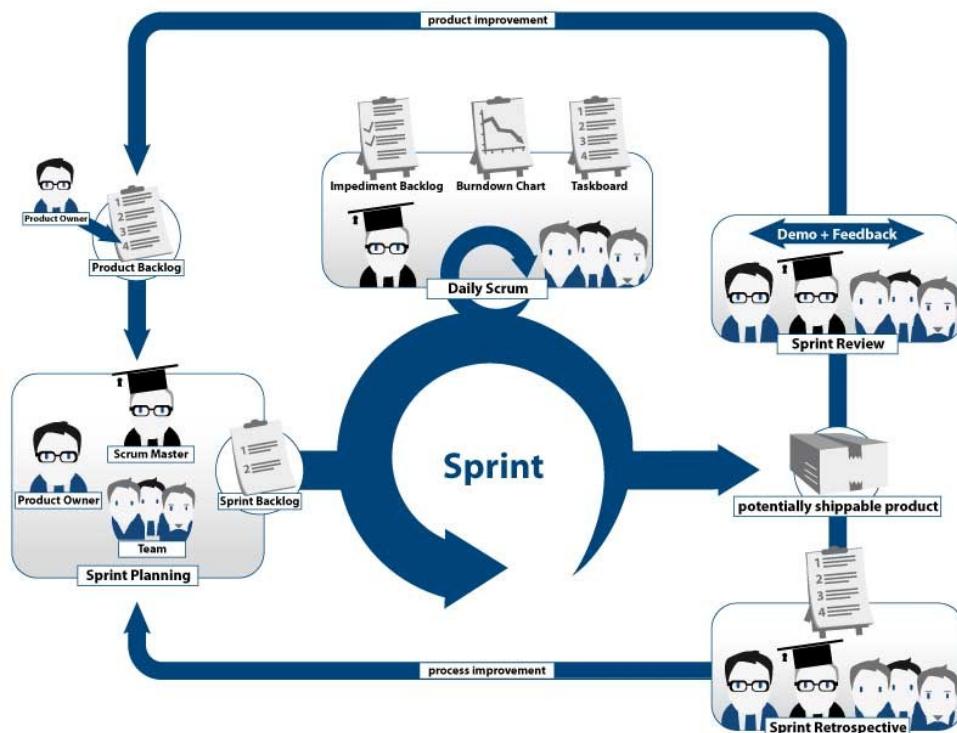
(SINESP – 2015) No Scrum, o evento que ocorre no final da sprint que serve para a equipe examinar a sprint passada e planejar melhorias é conhecido como:

- a) retrospectiva da sprint.
- b) avaliação da sprint.
- c) lições aprendidas da sprint.
- d) melhoria da sprint.
- e) fechamento da sprint.

**Comentários:** o evento que ocorre no final da sprint que serve para a equipe examinar a sprint passada e planejar melhorias é a retrospectiva da sprint (Letra A).

**Vamos fazer um resumão de tudo agora! Notem na imagem que tudo começa no canto superior esquerdo.** O Product Owner define o Product Backlog, isto é, uma lista com tudo que ele deseja que tenha em seu projeto. Então, os integrantes da Equipe Scrum fazem a Reunião de Planejamento e constroem a Sprint Backlog. O trabalho da sprint segue com reuniões diárias realizadas pelos desenvolvedores e atualizando os artefatos.

Ao final do sprint, há uma Revisão da Sprint – responsável por analisar se o incremento do produto entregue realmente satisfaz às expectativas dos clientes. Em seguida, realiza-se a última cerimônia, também conhecida como Retrospectiva da Sprint! Esse evento é responsável por analisar se o processo foi efetivamente utilizado e se há alguma sugestão de melhoria. **Por fim, essas melhorias servem de entrada para a próxima reunião de planejamento. Fechou?**



## Novidades do Scrum

# IMPORTANTE

O Scrum passou por melhorias desde sua versão inicial – acrescentando, modificando ou retirando conceitos. No entanto, infelizmente as bancas de concurso raramente explicitam no edital qual versão será cobrada em prova. Dessa forma, professores e alunos ficam vendidos. **De toda forma, essa aula foi feita baseada no Scrum 2013 com atualizações do Scrum 2017 e Scrum 2020, mas seguem abaixo resumidamente as melhorias da última versão.**

### Scrum 2020

#### SIMPLIFICAÇÃO GERAL DO GUIA

O Scrum 2020 coloca ênfase na eliminação de informações redundantes e complexas – assim como a remoção de qualquer referência remanescente em relação a tecnologia da informação (Ex: testes, sistemas, design, requerimento, etc), uma vez que ele pode ser utilizado para projetos de quaisquer áreas. O Guia Scrum agora possui menos do que 13 páginas e apresenta uma linguagem mais simplificada e compreensível para outros públicos.

#### DEFINIÇÃO DE SCRUM

Enquanto o Scrum 2017 se referia apenas a pessoas para resolução de problemas complexos, a nova versão trata também de times e organizações. Vejamos:

*Scrum é um framework leve que ajuda pessoas, times e organizações a gerar valor por meio de soluções adaptativas para problemas complexos.*

#### FIM DAS TRÊS PERGUNTAS

Removeram as famosas três perguntas da Reunião Diária, que eram um exemplo de perguntas que poderiam ser utilizadas pelo time, mas que para muitos virou regra para dar o status do projeto. O motivo da retirada foi que as perguntas nunca foram obrigatórias e muito menos uma forte sugestão para que guiassem todas as reuniões diárias – era simplesmente um exemplo que os autores utilizaram e que passou a ser visto e aplicado como regra.

Por essa razão, eles resolveram remover as perguntas por completo para que não fossem mais interpretadas incorretamente e gerassem uma disfunção nos Times Scrum.



## ARTEFATOS E COMPROMISSOS

Surgiram novas definições para artefatos, incluindo o conceito de compromissos conforme podemos ver a seguir:

*Os artefatos do Scrum representam trabalho ou valor. Eles são projetados para maximizar a transparência das principais informações. Assim, todos os que os inspecionam têm a mesma base para adaptação. Cada artefato contém um compromisso para garantir que ele forneça informações que aumentem a transparência e o foco contra o qual o progresso pode ser medido:*

- Para o Product Backlog, é a meta do produto;
- Para o Sprint Backlog, é a meta da sprint;
- Para o incremento, é a Definição de Pronto.

*Esses compromissos existem para reforçar o empirismo e os valores Scrum para o Scrum Team, e seus stakeholders.*

A Meta da Sprint e a Definição de Pronto já existiam na versão do Guia Scrum 2017, mas era confuso e não se sabia ao certo se eram artefatos ou não. Nesta versão, os autores fizeram questão de reforçar que Meta da Sprint, Definição de Pronto e Meta do Produto são compromissos que o Time Scrum deve ter em relação aos seus trabalhos e as suas entregas a fim de trazer transparência em direção ao progresso de cada artefato e do produto como um todo.

**Compromisso: Meta do Produto** – descreve um estado futuro do produto que pode servir como um alvo para o Scrum Team planejar. A Meta do produto está no Product Backlog. O restante do Product Backlog emerge para definir “o que” cumprirá a Meta do Produto.

*Um produto é um veículo para entregar valor. Tem um limite claro, stakeholders conhecidos, usuários ou clientes bem definidos. Um produto pode ser um serviço, um produto físico ou algo mais abstrato. A Meta do Produto é o objetivo de longo prazo para o Scrum Team. Eles devem cumprir (ou abandonar) um objetivo antes de assumir o próximo.*

**Compromisso: Meta da Sprint** – é o único objetivo da Sprint. Embora a Meta da Sprint seja um compromisso dos desenvolvedores, esta fornece flexibilidade em termos do trabalho exato necessário para alcançá-la.

*A Meta da Sprint também cria coerência e foco, encorajando o Scrum Team a trabalhar junto ao invés de iniciativas separadas. A Meta da Sprint é criada durante o evento Sprint Planning e então adicionada ao Sprint Backlog. Conforme os desenvolvedores trabalham durante a Sprint, eles mantêm a Meta da Sprint em mente. Se o trabalho acabar sendo diferente do que eles esperavam, eles colaboram com o Product Owner para negociar o escopo do Sprint Backlog dentro da Sprint sem afetar a Meta da Sprint.*

**Compromisso: Definição de Pronto** – é uma descrição formal do estado do Incremento quando ela atende às medidas de qualidade exigidas para o produto. No momento em que um item do Product Backlog atende a Definição de Pronto, um incremento nasce.

*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 Product Backlog não atender à Definição de Pronto, ele não poderá ser liberado ou mesmo apresentado na Sprint Review. Em vez disso, ele retorna ao Product Backlog*



*para consideração futura. Se a Definição de Pronto para um incremento faz parte dos padrões da organização, todos os Scrum Teams devem segui-la como mínimo.*

*Se não for um padrão organizacional, o Scrum Team deve criar uma Definição de Pronto apropriada para o produto. Os desenvolvedores devem estar em conformidade com a Definição de Pronto. Se houver vários Scrum Teams trabalhando juntos em um produto, eles devem definir e cumprir mutuamente a mesma Definição de Pronto.*

### **REMOÇÃO DO TERMO ‘TIME DE DESENVOLVIMENTO’**

A presença de um Time de Desenvolvimento passava a impressão de existir um sub-time no Time Scrum. Foi substituído pela responsabilidade Desenvolvedor. Isso reforça a ideia de termos apenas um único time: o Time Scrum, formado pelo Scrum Master, o Product Owner e os Desenvolvedores – seus objetivos devem ser o mesmo.

A mudança não é apenas semântica: ao remover o conceito de sub-time dentro da equipe e deixar claro que todas essas pessoas pertencem ao mesmo time, o Time Scrum, isso cria um compromisso mais forte entre todos para a entrega da Meta da Sprint. O Time Scrum é apenas uma equipe com três responsabilidades diferentes.

*A unidade fundamental do Scrum é um pequeno time de pessoas, um Scrum Team. O Scrum Team consiste em um Scrum Master, um Product owner e Developers. Dentro de um Scrum Team, não há sub-times ou hierarquias. É uma unidade coesa de profissionais focados em um objetivo de cada vez, a Meta do Produto.*

### **TAMANHO DA EQUIPE**

Na versão do Guia Scrum 2017, estava descrito que o Time de Desenvolvimento deveria idealmente ser composto por 3 a 9 integrantes. Com o objetivo de se tornar ainda menos prescritivo, agora não há tamanho mínimo ou máximo. No entanto, o Scrum Guide comenta que o Scrum Team normalmente tem 10 ou menos integrantes – incluindo na conta o Scrum Master e o Product Owner.

*O Scrum Team é pequeno o suficiente para permanecer ágil e grande o suficiente para concluir um trabalho significativo dentro de uma Sprint, normalmente 10 ou menos pessoas. Em geral, descobrimos que times menores se comunicam melhor e são mais produtivos. Se os Scrum Teams se tornarem muito grandes, eles devem considerar a reorganização em vários Scrum Teams coesos, cada um focado no mesmo produto. Portanto, eles devem compartilhar a mesma meta do produto, Product Backlog e Product Owner.*

### **“POR QUE” DO PLANEJAMENTO DA SPRINT**

A Sprint Planning trazia os tópicos “O que” e “Como” e, na versão 2020, um novo tópico foi adicionado: “Por que”. Logo, deve ser respondido “Por que esta Sprint é valiosa?” e “Por que esta deve ser realizada?”. Muitas vezes, sabemos o que fazer, como fazer, mas não sabemos por que fazer – o que acaba gerando produtos inúteis e promovendo o desperdício (o que não é preconizado pelo lean thinking) por toda a organização.



## AUTO-ORGANIZÁVEL → AUTO-GERENCIÁVEL

Houve uma mudança de termos: saiu o “auto-organizável” e entrou o “auto-gerenciável”. Neste ponto, os autores do guia oficial quiseram passar uma forte mensagem em relação a autonomia e responsabilidades do Scrum Team no sentido de que ele não é só responsável por escolher quem e como farão o trabalho da sprint, mas devem ser auto-gerenciados de modo a escolher quem, como e no que trabalhar.

O Scrum Team deve ter as responsabilidades compartilhadas de selecionar, entender e priorizar no que vão trabalhar, quem irá trabalhar e como irão trabalhar para cumprir os compromissos.

## PRODUCT OWNER COMPARTILHADO

Um incremento de texto importante foi o que o Product Owner pode ser compartilhado com múltiplos times. Se os Scrum Teams se tornarem muito grandes, eles devem considerar a reorganização em vários times menores e coesos, cada um focado no mesmo produto e devem compartilhar a mesma Meta do Produto, Product Backlog e Product Owner. No entanto, apenas um PO deve ser considerado, além do mesmo Backlog do Produto e Meta do Produto.



## RESUMO

### [Guia Scrum - Versão 2017]

Scrum: um framework dentro do qual pessoas podem tratar e resolver problemas complexos e adaptativos, enquanto produtiva e criativamente entregam produtos com o mais alto valor possível. Scrum é: leve, simples de entender e difícil de dominar.

### [Guia Scrum - Versão 2020]

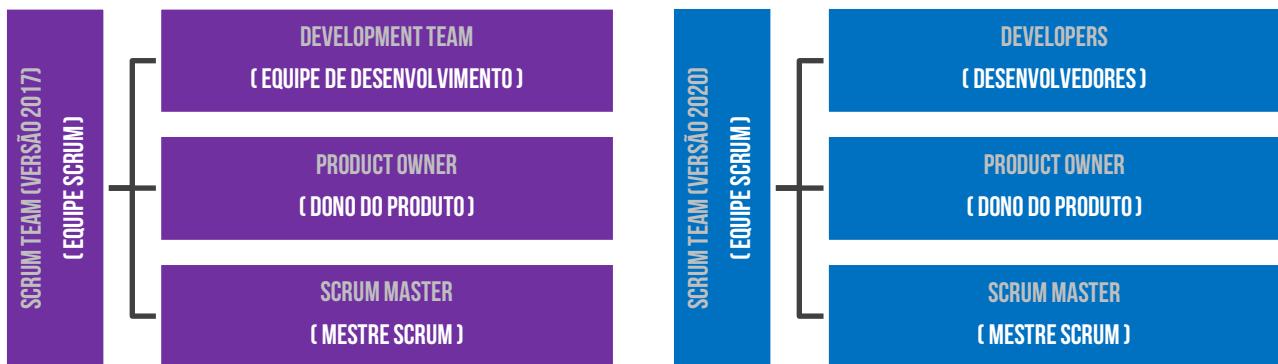
Scrum é um framework leve que ajuda pessoas, times e organizações a gerar valor por meio de soluções adaptativas para problemas complexos.



PILARES	DESCRIÇÃO
<b>TRANSPARÊNCIA</b>	Todo trabalho deve ser claramente definido e conhecido por todas as partes envolvidas no projeto.
<b>INSPEÇÃO</b>	Todo trabalho deve ser inspecionado com a frequência necessária para garantir a qualidade do produto.
<b>ADAPTAÇÃO</b>	O projeto deve ser capaz de se adaptar o projeto às necessidades de negócios.



VALORES	DESCRIÇÃO
CORAGEM	Os integrantes de um projeto precisam ter coragem para fazer a coisa certa e trabalharem juntos removendo impedimentos, buscando soluções.
FOCO	Os integrantes de um projeto precisam focar no trabalho durante a sprint e nas metas designadas – time disperso perde produtividade e não alcança os objetivos.
COMPROMETIMENTO	Os integrantes se comprometem com o trabalho que se responsabilizou em fazer, envolvendo-se e não abandonando pela metade ou entregando sem qualidade.
RESPEITO	Os integrantes se respeitam entre si a fim de manter a colaboração, a integração e o bom ambiente de trabalho.
ABERTURA	Os integrantes devem poder ser francos, expor ideias e propostas mesmo que elas não sejam proveitosas. Momentos de debates, discussões e sugestões são ideais.



Responsável pela gestão de pessoas e gestão do processo.

Ele deve garantir que o Scrum seja entendido e aplicado. O Scrum Master faz isso para garantir que a Equipe Scrum adere à teoria, práticas e regras do Scrum.

O Scrum Master ajuda aqueles que estão fora da Equipe Scrum a entender quais as suas interações com a Equipe Scrum são úteis e quais não são.

O Scrum Master ajuda todos a mudarem estas interações para maximizar o valor criado pela Equipe Scrum.

Ele é responsável por orientar o Product Owner na criação e ordenação do Product Backlog.

Ele é responsável por garantir que as regras do Scrum estejam sendo cumpridas e seus valores estejam sendo seguidos.

Ele é responsável por ajudar a remover impedimentos que o time encontre, fazendo isso sem o uso de qualquer autoridade.

Ele utiliza técnicas de facilitação e coaching para que os membros do time consigam visualizar os problemas e encontrem a melhor solução.

Durante eventos, ele é responsável por fazer com que a reunião flua adequadamente, utilizando técnicas de facilitação, embora não seja o responsável pela condução.

Ele ajuda a treinar os desenvolvedores em autogerenciamento e interdisciplinaridade.

Ele treina os desenvolvedores em ambientes organizacionais nos quais o Scrum não é totalmente adotado e compreendido.

Ele ensina a Equipe Scrum a criar itens do Product Backlog de forma clara e concisa.

## RESPONSABILIDADES SCRUM MASTER (SM)

Ele comunica claramente a visão, objetivo e itens do Product Backlog para os desenvolvedores.

**RESPONSABILIDADES  
DESENVOLVEDORES**

Responsável pela micro-gestão e pela criação do produto.

Eles são auto-organizados. Ninguém (nem mesmo o SM) diz aos desenvolvedores como transformar o Product Backlog em incrementos de funcionalidades potencialmente utilizáveis.

Times de Desenvolvimento são multifuncionais, possuindo todas as habilidades necessárias, enquanto equipe, para criar o incremento do Produto.

O Scrum não reconhece títulos específicos para os desenvolvedores, independentemente do trabalho que está sendo realizado pela pessoa;

Individualmente, os desenvolvedores podem ter habilidades especializadas, mas a responsabilidade pertence aos desenvolvedores como um todo.

Os desenvolvedores não contêm sub-times dedicados a domínios específicos de conhecimento, tais como teste ou análise de negócios.

Os desenvolvedores são estruturados e autorizados pela organização para organizar e gerenciar seu próprio trabalho.

**RESPONSABILIDADES  
PRODUCT OWNER (PO)**

Ele é responsável pela macro-gestão e pela gestão do produto.

Ele é o responsável por maximizar o valor do produto e do trabalho dos desenvolvedores, sendo o único que pode gerenciar o Product Backlog.

Ele pode até delegar as atividades de gerenciamento para os desenvolvedores, mas ainda será considerado o responsável pelos trabalhos.

Ele é responsável por priorizar/ordenar os itens do Product Backlog e seleciona aqueles que serão implementados.

Ele é responsável por garantir o ROI (Return On Investment ou Retorno sobre Investimento).

Ele é responsável por expressar claramente os itens do Product Backlog.

Ele é responsável por garantir que o Backlog do Produto seja visível, transparente, claro para todos, e mostrar o que a Equipe Scrum vai trabalhar a seguir.

Ele é responsável por garantir que os desenvolvedores entendam os itens do Product Backlog no nível necessário.

**PRODUCT BACKLOG**

Trata-se de uma lista ordenada (por valor, risco, prioridade, entre outros) de requisitos ou funcionalidades que o produto deve conter criada pela Equipe Scrum e gerenciada pelo Product Owner.

**SPRINT BACKLOG**

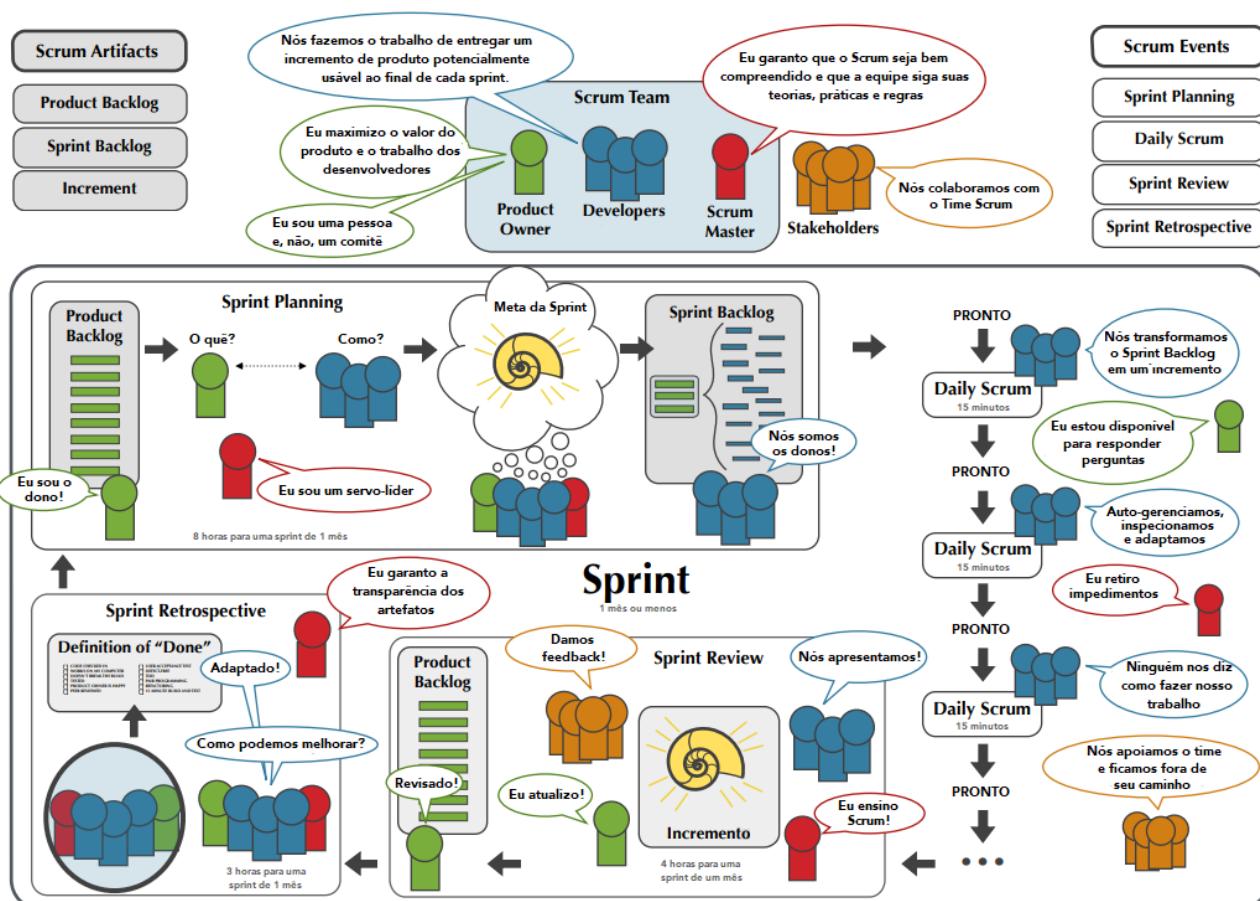
Trata-se de conjunto de itens selecionados do Product Backlog, mais a meta da sprint e mais um plano de ação para entregar um incremento potencialmente usável – é criado e gerenciado pelos desenvolvedores.

**SPRINT REVIEW**

Trata-se da soma de todos os itens do Backlog do Produto completados durante a Sprint e o valor dos incrementos de todas as sprints anteriores – sendo validado como “pronto”.



<b>DEFINIÇÃO DE READY</b>	Conjunto de critérios que indicam que já existem informações suficientes para um requisito começar a ser desenvolvido.
<b>DEFINIÇÃO DE DONE</b>	Conjunto de critérios que indicam que uma determinada história de usuário atende a todos os requisitos de aceitação para se tornar um incremento.
<b>SPRINT PLANNING</b>	Reunião dividida em duas partes que possui duração de até 8 horas. Na primeira parte, a equipe seleciona, alinha e detalha os itens que vão ser desenvolvidos na próxima sprint. Na segunda parte, cada item é estimado e decomposto nas tarefas necessárias para produzir as entregas.
<b>DAILY SCRUM</b>	Reunião diária para alinhar a comunicação do projeto, inspecionar o progresso para a meta, identificar impedimentos e adaptar o backlog da sprint, se necessário. Não pode ter mais que 15 minutos de duração, ocorrendo sempre no mesmo local e horário.
<b>SPRINT REVIEW</b>	Reunião de até 4h de duração realizada ao final de cada sprint para apresentar ao Product Owner as funcionalidades implementadas para que ele possa validá-las e eventualmente adaptar futuras modificações. Trata-se de um evento informal para apresentação do incremento e colaboração sobre os próprios passos.
<b>SPRINT RETROSPECTIVE</b>	Reunião de até 3h de duração realizada após a Sprint Review. No entanto, em vez de validar o produto, a equipe busca revisar e validar o processo executado para gerar as funcionalidades. A ideia é planejar maneiras de aumentar a qualidade e efetividade do processo.



PARA MAIS DICAS:

[WWW.INSTAGRAM.COM/PROFESSORDIEGOCARVALHO](http://WWW.INSTAGRAM.COM/PROFESSORDIEGOCARVALHO)

