



A Lightweight Guide to the Theory and Practice of Scrum Version 2.0

Pete Deemer
GoodAgile
www.goodagile.com

Gabrielle Benefield
Evolve
www.evolvebeyond.com

Craig Larman
www.craiglarman.com

Bas Vodde
Odd-e
www.odd-e.com



InfoQ
ENTERPRISE SOFTWARE
DEVELOPMENT SERIES

Uma nota aos leitores: há muitas descrições concisas do Scrum disponíveis online, e esta cartilha tem como objetivo fornecer o próximo nível de detalhe sobre as práticas. Não se destina a ser a etapa final na educação Scrum; as equipes que estão considerando adotar o Scrum são aconselhadas a se equiparem com o conhecimento de Ken Schwaber *Gerenciamento Ágil de Projetos com Scrum* ou, Mike Cohn's *Sucesso com Ágil* e aproveite as muitas opções excelentes de treinamento e coaching Scrum disponíveis; detalhes completos estão em scrumalliance.org. Nossos agradecimentos a Ken Schwaber, Dr. Jeff Sutherland e Mike Cohn por suas generosas contribuições.

A versão mais recente do Primer pode ser encontrada em: http://www.infoq.com/minibooks/Scrum_Primer

As traduções podem ser encontradas em: <http://www.scrumprimer.org/>

© 2012 Pete Deemer, Gabrielle Benefield, Craig Larman, Bas Vodde

Além do Desenvolvimento Tradicional

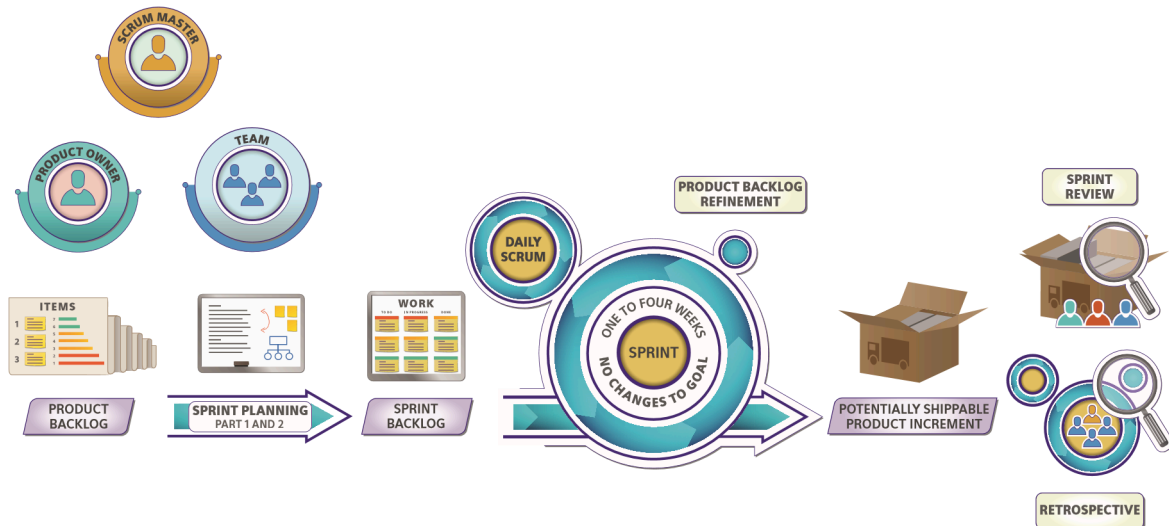
O desenvolvimento tradicional com grupos de função única, ciclos de feedback atrasados ou fracos, planejamento preditivo antecipado e um fluxo sequencial da análise ao teste não tem muito sucesso no mundo volátil de hoje. Esta abordagem atrasa o feedback, a aprendizagem e o potencial retorno do investimento devido à ausência de software funcional real até o final do jogo, causando falta de transparência, falta de capacidade de melhorar, redução na flexibilidade e um aumento nos riscos comerciais e técnicos.

Uma alternativa – equipes multifuncionais com desenvolvimento iterativo – também existe há décadas, mas não era tão amplamente utilizada quanto o modelo tradicional.

O Scrum empacota conceitos comprovados de desenvolvimento de produtos em uma estrutura simples, incluindo: equipes reais, equipes multifuncionais, equipes autogerenciadas, ciclos iterativos curtos de feedback de ciclo completo e redução do custo da mudança. Esses conceitos aumentam a agilidade e o feedback, permitem um ROI mais rápido e reduzem os riscos.

Visão geral

Scrum é uma estrutura de desenvolvimento na qual equipes multifuncionais desenvolvem produtos ou projetos de maneira iterativa e incremental. Estrutura o desenvolvimento em ciclos de trabalho chamados **Sprints**. Essas iterações não duram mais do que quatro semanas cada (o mais comum é de duas semanas) e ocorrem uma após a outra sem pausa. As Sprints são *com limite de tempo(timeboxed)* – terminam numa data específica, quer a obra tenha sido concluída ou não, e *nunca são estendidas*. Normalmente os Times Scrum escolhem uma duração de Sprint e a usam para todos os seus Sprints até que melhorem e possam usar um ciclo mais curto. No início de cada Sprint, um *Cross functional Team* (de cerca de sete pessoas) seleciona **Items** (requisitos do cliente) a partir de uma lista priorizada. A Equipe concorda com uma meta coletiva do que eles acreditam que podem entregar até o final do Sprint, algo que seja tangível e que será verdadeiramente “feito”(Done). Durante a Sprint não poderão ser adicionados novos itens; Scrum abraça a mudança para o *próximo* Sprint, mas o atual Sprint curto tem como objetivo focar em uma meta pequena, clara e relativamente estável. Todos os dias a Equipe se reúne brevemente para inspecionar seu progresso e ajustar os próximos passos necessários para concluir o trabalho restante. No final do Sprint, a Equipe analisa o Sprint com as partes interessadas e demonstra o que foi construído. As pessoas obtêm feedback que pode ser incorporado no próximo Sprint. Scrum enfatiza o produto funcional no final do Sprint que está realmente “pronto”(Done); no caso de software, isso significa um sistema integrado, totalmente testado, documentado pelo usuário final e potencialmente distribuível. As principais funções, artefatos e eventos estão resumidos na Figura 1.



SCRUM

Figura 1. Visão geral do Scrum

Um tema importante no Scrum é “inspecionar e adaptar”(inspect and adapt). Como o desenvolvimento envolve inevitavelmente aprendizagem, inovação e surpresas, o Scrum enfatiza dar um pequeno passo no desenvolvimento, inspecionando tanto o produto resultante quanto a eficácia das práticas atuais e, em seguida, adaptando os objetivos do produto e as práticas do processo. *Repita para sempre.*

Funções do Scrum

No Scrum, existem três funções: Dono do Produto(PO-Product Owner),Equipe (Team) e Scrum Master (SM). Juntos, eles são conhecidos como Equipe Scrum (Scrum Team).

O **Proprietário do produto(PO)** é responsável por maximizar o retorno sobre o investimento (ROI), identificando características do produto, traduzindo-as em uma lista priorizada, decidindo quais devem estar no topo da lista para o próximo Sprint e continuamente repriorizando e refinando a lista. O Dono do Produto é responsável pelos lucros e perdas do produto, presumindo que seja um produto comercial. No caso de uma aplicação interna, o Product Owner não é responsável pelo ROI no sentido de um produto comercial (que irá gerar receita), mas ainda é responsável por maximizar o ROI no sentido de escolher – a cada Sprint – o maior itens de valor. Na prática, “valor” é um termo confuso e a priorização pode ser influenciada pelo desejo de satisfazer clientes-chave, alinhamento com objetivos estratégicos, ataque a riscos, melhoria e outros fatores. Em alguns casos, o Product Owner e o cliente são a mesma pessoa; isso é comum para aplicativos internos. Em outros, o cliente pode ser milhões de pessoas com diversas necessidades; nesse caso, a função de Proprietário do Produto é semelhante à posição de Gerente de Produto ou Gerente de Marketing de Produto em muitas organizações de produtos. No entanto, o Product Owner é um pouco diferente de um Gerente de Produto tradicional porque interage ativa e regularmente com a Equipe, prioriza trabalhando com todas as partes interessadas e revisando os resultados a cada Sprint, em vez de delegar as decisões de desenvolvimento a um gerente de projeto. É importante notar que no Scrum existe uma e apenas uma pessoa que atua como – e tem a autoridade final de – Product Owner, e ela é responsável pelo valor do trabalho; embora essa pessoa não precise trabalhar sozinha.

A **Equipe (também chamada de Equipe de Desenvolvimento)** constrói o produto que o Product Owner indica: a aplicação ou site, por exemplo. A Equipe no Scrum é “multifuncional” – inclui todo o conhecimento necessário para entregar o produto potencialmente entregável a cada Sprint – e é “auto-organizada” (autogerenciada), com um alto grau de autonomia e responsabilidade. A equipe decide quantos itens (do conjunto oferecido pelo Product Owner) serão construídos em um Sprint e qual a melhor forma de atingir esse objetivo.

Cada membro da equipe é apenas um *membro da equipe*. Observe que não há títulos fixos de especialistas em um grupo que adota Scrum; não há analista de negócios, nem DBA, nem arquiteto, nem líder de equipe, nem designer de interação/UX, nem programador. Eles trabalham juntos durante cada Sprint da maneira que for apropriada para atingir a meta que estabeleceram para si mesmos.

Como existem apenas *membros do time* (*Team members*), a equipe não é apenas multifuncional, mas também demonstra *multi aprendizagem*: cada pessoa certamente tem qualidades especiais, mas também continua aprendendo outras especialidades. Cada pessoa terá competências primárias, secundárias e até terciárias, e deverá “ir até onde está o trabalho”; os indivíduos realizam tarefas em áreas menos familiares para ajudar a completar um item. Por exemplo, uma pessoa cuja habilidade principal é design de interação poderia ter uma habilidade secundária em testes automatizados; alguém com habilidades primárias em redação técnica também pode ajudar na análise e programação.

A Equipe no Scrum é formada por sete pessoas ou menos, e para um produto de software a Equipe pode incluir pessoas com habilidades em análise, desenvolvimento, testes, design de interface, design de banco de dados, arquitetura, documentação e assim por diante. A equipe desenvolve o produto e fornece ideias ao Product Owner sobre como tornar o produto excelente. No Scrum, as equipes são mais produtivas e eficazes se todos os membros estiverem 100% dedicados ao trabalho de um produto durante o Sprint; a equipe evita realizar múltiplas tarefas em vários produtos ou projetos, para evitar o desperdício dispendioso de atenções divididas e mudança de contexto. Equipes estáveis estão associadas a maior produtividade, portanto evite trocar de membro da equipe. Grupos de produtos com muitas pessoas são organizados em diversas equipes, cada uma focada em diferentes recursos do produto, com estreita coordenação de seus esforços. Como uma equipe geralmente faz todo o trabalho (planejamento, análise, programação e testes) para um recurso completo centrado no cliente, as equipes também são conhecidas como *equipes de recursos* (*features teams*).

O **Scrum Master** ajuda o grupo de produtos a aprender e aplicar o Scrum para obter valor comercial. O Scrum Master faz tudo o que está ao seu alcance para ajudar a equipe, o Product Owner e a organização a terem sucesso. O Scrum Master *não* é o gerente dos membros da equipe, nem é gerente de projeto, líder de equipe ou representante da equipe. Em vez disso, o Scrum Master *serve* o time; ele ou ela ajudam a remover impedimentos, protege a Equipe de interferências externas e ajuda a Equipe a adotar práticas modernas de desenvolvimento. Ele ou ela educa, treina e orienta o Product Owner, a equipe e o resto da organização no uso habilidoso do Scrum. O Scrum Master é um *treinador* e *professor*. O Scrum Master garante que todos (incluindo o Dono do Produto e aqueles na gestão) entendam os princípios e práticas do Scrum e ajudem a liderar a organização através das mudanças muitas vezes difíceis necessárias para alcançar o sucesso com o desenvolvimento ágil. Como o Scrum torna visíveis muitos impedimentos e ameaças à eficácia do Time e do Product Owner, é importante ter um Scrum Master engajado trabalhando energeticamente para ajudar a resolver esses problemas, ou o Time ou o Product Owner acharão difícil ter sucesso. Deve haver um Scrum Master dedicado em tempo integral, embora uma equipe menor possa ter um membro da equipe desempenhando essa função (carregando uma carga mais leve de trabalho regular quando o faz). Grandes Scrum Masters podem vir de qualquer formação ou disciplina: Engenharia, Design, Testes, Gerenciamento de Produtos, Gerenciamento de Projetos ou Gerenciamento de Qualidade.

O Scrum Master e o Product Owner *não* podem ser o mesmo indivíduo, pois seus focos são muito diferentes e combiná-los muitas vezes leva a confusão e conflito. Um resultado infeliz comum da combinação dessas funções é um Dono do Produto micro gerenciado, que é o oposto das equipes autogerenciadas que o Scrum exige. Ao contrário de um gestor tradicional, o Scrum Master não diz às pessoas o que fazer nem atribui tarefas – eles facilitam o processo, apoiando a Equipe à medida que esta se organiza e gere. Se o Scrum Master estava anteriormente em uma posição de gerenciamento do Time, ele precisará mudar significativamente sua mentalidade e estilo de interação para que o Time tenha sucesso com o Scrum.

Nota: não existe nenhuma função de gerente de projeto no Scrum. Isso ocorre porque nada é necessário; as responsabilidades tradicionais de um gerente de projeto foram divididas e redistribuídas entre as três funções do Scrum, e principalmente para a Equipe e o Dono do Produto, em vez do Scrum Master. Praticar Scrum com a adição de um gerente de projeto indica um mal-entendido fundamental do Scrum e normalmente resulta em responsabilidades conflitantes, autoridade pouco clara e resultados abaixo do ideal. Às vezes, um (ex-)gerente de projeto pode assumir o papel de Scrum

Master, mas o sucesso dessa abordagem depende muito do indivíduo e de quão bem ele entende a diferença fundamental entre as duas funções, tanto no dia a dia, responsabilidades e na mentalidade necessária para ter sucesso. Uma boa maneira de compreender completamente o papel do Scrum Master e começar a desenvolver as competências essenciais necessárias para o sucesso é frequentar o treinamento Certified Scrum Master da Scrum Alliance.

Além destas três funções, existem outras partes interessadas que contribuem para o sucesso do produto, incluindo gestores, clientes e utilizadores finais. Algumas partes interessadas, como gerentes funcionais (por exemplo, um gerente de engenharia), podem encontrar mudanças em seu papel, embora ainda valiosas, ao adotar o Scrum. Por exemplo:

- eles apoiam a equipe respeitando as regras e o espírito do Scrum
- eles ajudam a remover impedimentos que a equipe e o proprietário do produto identificam
- eles disponibilizam seus conhecimentos e experiência

No Scrum, esses indivíduos substituem o tempo que antes passavam desempenhando o papel de “babá” (atribuindo tarefas, obtendo relatórios de status e outras formas de microgerenciamento) por tempo como “guru” e “servo” do Time (mentoria, coaching, ajudando a remover obstáculos, ajudando a resolver problemas, fornecendo informações criativas e orientando o desenvolvimento de habilidades dos membros da equipe). Nesta mudança, os gestores poderão necessitar de alterar o seu estilo de gestão; por exemplo, utilizar o questionamento socrático para ajudar a Equipe a descobrir a solução para um problema, em vez de simplesmente decidir uma solução e atribuí-la à Equipe.

Backlog do produto(Product Backlog)

Quando um grupo está planejando a transição para o Scrum, antes que o primeiro Sprint possa começar, eles precisam de um **Backlog do produto**, uma lista priorizada (ordenada 1, 2, 3,...) de recursos centrados no cliente.

O Product Backlog existe (e evolui) ao longo da vida do produto; é o roteiro do produto (**Figura 2 e Figura 3**). A qualquer momento, o Product Backlog é a visão única e definitiva de “tudo o que poderia ser feito pelo Time, em ordem de prioridade”. Existe apenas *um* único Product Backlog para *um* produto; isso significa que o Dono do Produto é obrigado a tomar decisões de priorização em todo o espectro, representando os interesses das partes interessadas (incluindo a Equipe).

	Novas estimativas na Sprint ...								
Prioridade	Item	Detalhes (wiki URL)	Estimativa de tamanho inicial	1	2	3	4	5	6
1	Como comprador, quero colocar um livro em um carrinho de compras (veja os esboços da IU na página wiki)	...	5						
2	Como comprador, quero remover um livro do carrinho de compras	...	2						
3	Melhore o desempenho do processamento de transações (veja as métricas de desempenho alvo no wiki)	...	13						
4	Investigue soluções para acelerar a validação de cartão de crédito (veja as métricas de desempenho alvo na wiki)	...	20						
5	Atualize todos os servidores para Apache 2.2.3	...	13						
6	Diagnosticar e corrigir erros de script de processamento de pedidos (Com Bugz ID 14823)	...	3						
7	Como comprador, quero criar e salvar uma lista de desejos	...	40						
8	Como comprador, quero adicionar ou excluir itens da minha lista de desejos	...	20						

Figura 2. O Backlog do Produto

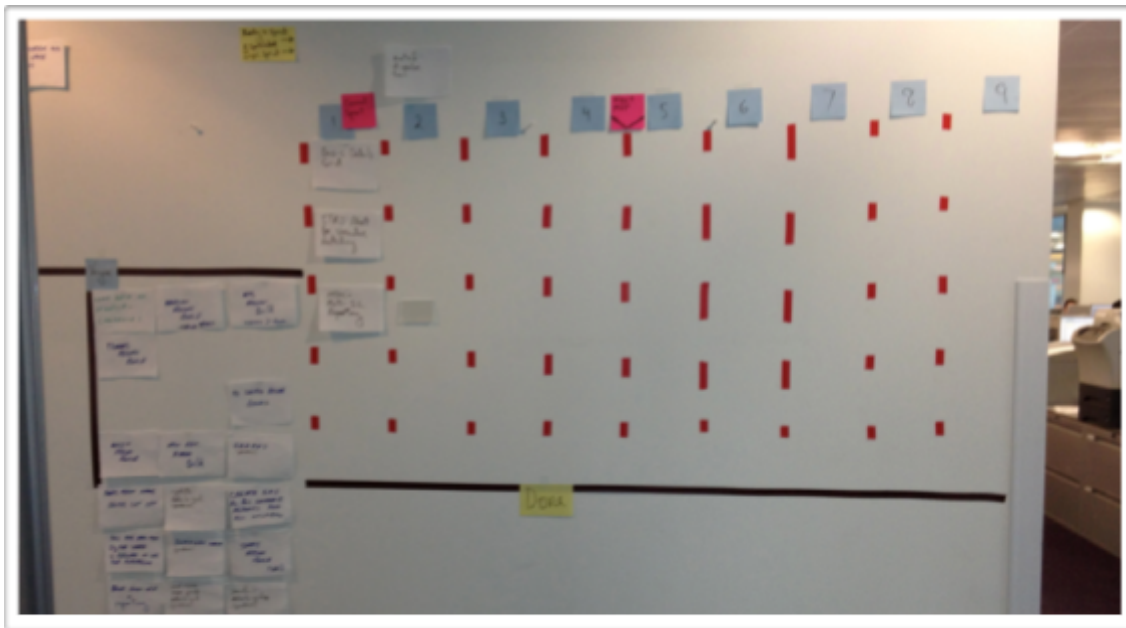


Figura 3. Gestão Visual: Itens do Backlog do Produto na parede

O Backlog do Produto inclui uma variedade de **Items**, principalmente novos recursos para o cliente (“permitir que todos os usuários coloquem livros no carrinho de compras”), mas também e *principalmente* metas de melhoria de engenharia (por exemplo, “reescrever o sistema de C++ para Java”), metas de melhoria (por exemplo, “acelerar nossos testes”), trabalho de pesquisa (“investigar soluções para acelerar a validação de cartão de crédito”) e, possivelmente, defeitos conhecidos (“diagnosticar e corrigir erros de script de processamento de pedidos”) se houver apenas alguns problemas. (Um sistema com muitos defeitos geralmente possui um sistema de rastreamento de defeitos separado.)

Os itens do Backlog do Produto são articulados de forma clara e sustentável. Ao contrário do mal-entendido popular, o Product Backlog *não* contém somente “histórias de usuários”; simplesmente contém *Items*. Esses itens podem ser expressos como histórias de usuários, casos de uso ou qualquer outra abordagem de requisitos que o grupo considere útil. Mas seja qual for a abordagem, a maioria dos itens deve focar na entrega de valor aos clientes.

Um bom Backlog de Produto é PROFUNDO...

Detalhado adequadamente. Os itens de prioridade máxima são mais refinados e detalhados do que os itens de prioridade mais baixa, uma vez que os primeiros serão trabalhados mais cedo do que os últimos. Por exemplo, os 10% principais do backlog podem ser compostos por itens muito pequenos e bem analisados, e os outros 90%, muito menos.

Estimado. Os itens da versão atual precisam ter estimativas e, além disso, devem ser considerados para serem reestimados a cada Sprint à medida que todos aprendem e novas informações surgem. A equipe fornece ao Dono do Produto o *esforço* estimado para cada item do Product Backlog, e talvez também *risco técnico* estimado. O Product Owner e outras partes interessadas do negócio fornecem informações sobre o valor do produto solicitações, que podem incluir receitas obtidas, custos reduzidos, riscos comerciais, importância para várias partes interessadas e muito mais.

Emergente. Em resposta ao aprendizado e à variabilidade, o Backlog do Produto é refinado regularmente. A cada Sprint, itens podem ser adicionados, removidos, modificados, divididos e alterados em prioridade. Assim, o Product Backlog é continuamente atualizado pelo Product Owner para refletir mudanças nas necessidades do cliente, novas idéias ou insights, movimentos da concorrência, obstáculos técnicos que aparecem e assim por diante.

Priorizado. Os itens no topo do Product Backlog são priorizados ou *encomendados* em uma ordem 1-N. Em geral, os itens de maior prioridade devem entregar os resultados mais *vale o seu dinheiro*: muito

retorno (valor comercial) por baixo investimento (custo). Outra motivação para aumentar a prioridade de um item é *enfrente riscos elevados antecipadamente, antes que os riscos ataquem você*.

O desenvolvimento tradicional geralmente não enfatiza a entrega de acordo com os mais elevados *vale o seu dinheiro*, mas este é um tema do Scrum e, portanto, o Product Owner precisará aprender como avaliar o impacto do “valor comercial”. Isso é algo que o Scrum Master pode ajudar o Product Owner a aprender. O que significa “valor comercial”? Alguns grupos de produtos usam uma estimativa simples de ponto de valor relativo para cada item do Product Backlog que sintetiza uma “estimativa” de fatores, incluindo ganho de receita, redução de custos, preferências das partes interessadas, diferenciação de mercado e assim por diante. Alguns financiam um item específico através de um ou mais clientes que pagam pelo seu desenvolvimento e, portanto, usam a receita exata (curto prazo) desse item como proxy do valor. Para outros grupos, essa estimativa de valor específica do item é muito desfocada ou granular; eles aplicam uma abordagem mais ampla baseada em resultados de negócios (“aumentar as assinaturas em 10% até 1º de setembro”), na qual o valor só é entregue quando vários itens que contribuem para os resultados são entregues juntos. Nesse caso, o Product Owner precisa definir o próximo incremento do Produto Mínimo Viável.

Para estimativas de esforço, uma técnica comum é estimar em termos de tamanho relativo (considerando esforço, complexidade e incerteza) usando uma unidade de “pontos de história” ou simplesmente “pontos”.

Estas são apenas sugestões; Scrum não define a técnica para expressar ou priorizar itens no Product Backlog e não define a técnica ou unidades de estimativa.

Uma técnica comum usada no Scrum é rastrear quanto trabalho é concluído em cada Sprint; por exemplo, uma média de 26 pontos concluídos por Sprint. Com essas informações eles podem projetar uma data de lançamento para completar todos os recursos, ou quantos recursos podem ser concluídos até uma data fixa, se a média continuar e nada mudar. Essa média é chamada de “velocidade”. A velocidade é expressa nas mesmas unidades que as estimativas de tamanho do item do Product Backlog.

Os itens do Product Backlog podem variar significativamente em tamanho ou esforço. Os maiores são divididos em itens menores durante o workshop de Refinamento do Backlog do Produto ou na Reunião de Planejamento do Sprint, e os menores podem ser consolidados. Os itens do Backlog do Produto para os próximos Sprints devem ser pequenos e refinados o suficiente para serem compreendidos pela Equipe, permitindo que as previsões feitas na reunião de Planejamento do Sprint sejam significativas; isso é chamado de tamanho “acionável”.

Grandes melhorias de engenharia que consomem muito tempo e dinheiro devem estar no Product Backlog, uma vez que podem ser um investimento comercial opcional, a ser feito em última análise pelo Product Owner orientado para os negócios. Observe que no Scrum, o Time tem autoridade independente sobre quantos itens do Backlog do Produto eles decidem levar para um Sprint, então eles são livres de forma independente para assumir pequenos trabalhos de melhoria de engenharia, pois podem ser considerados parte do custo normal de negócios e o que é necessário para um desenvolvedor fazer seu trabalho corretamente. Dito isto, em cada Sprint, a *maior* parte do tempo de uma equipe geralmente deve ser dedicada às metas do Proprietário do Produto, e não às tarefas internas de engenharia.

Um dos *mitos* sobre o Scrum é que ele impede você de escrever especificações detalhadas; na realidade, cabe ao Product Owner e à Equipe decidir quantos detalhes são necessários, e isso irá variar de um item do backlog para outro, dependendo da visão da Equipe e de outros fatores. Afirme o que é importante no menor espaço necessário – em outras palavras, não descreva todos os detalhes possíveis de um item, apenas deixe claro o que é necessário para que ele seja compreendido e aumente isso com um diálogo contínuo entre a Equipe e o Product Owner e partes interessadas. Itens do Backlog do Produto de baixa prioridade, que não serão trabalhados por algum tempo, geralmente são “grossos” (grandes, com requisitos menos detalhados, alto nível). Itens de backlog do produto de alta prioridade e granularidade que serão implementados em breve tendem a ter mais detalhes.

Definição de Pronto (Definition of Done)

A saída de cada Sprint é oficialmente chamada de Incremento de Produto Potencialmente Entregável. Antes de iniciar o primeiro Sprint, o Dono do Produto, a Equipe e o Scrum Master devem revisar *tudo* o que é necessário para que um item do Backlog do Produto seja potencialmente entregável. Todas as atividades necessárias para enviar o produto devem ser incluídas na definição de Potencialmente Entregável e, portanto, devem ser realizadas durante o Sprint.

Infelizmente, quando as equipes começam a usar o Scrum, muitas vezes não conseguem atingir a meta de entregar um incremento potencialmente distribuível a cada Sprint. Muitas vezes, isso ocorre porque a equipe carece de automação ou não é multifuncional o suficiente (por exemplo, os redatores técnicos ainda não estão incluídos na equipe multifuncional). Com o tempo, a equipe precisa melhorar para poder entregar um incremento de produto potencialmente entregue a cada Sprint, mas, para começar, precisará criar uma linha de base de suas capacidades existentes. Isso está registrado na Definição de Pronto.

Antes do primeiro Sprint, o Dono do Produto e a Equipe precisam chegar a um *acordo* sobre uma Definição de Pronto (Definition of Done-DoD), que é um subconjunto das atividades necessárias para criar um Incremento de Produto Potencialmente Entregável (para uma boa Equipe, será a mesma coisa). A Equipe planejará seu trabalho de Sprint de acordo com esta Definição de Pronto (DoD).

Um bom Product Owner sempre desejará que a Definição de Pronto seja o mais próximo possível do Potencialmente Entregável, pois isso aumentará a transparência no desenvolvimento e diminuirá *atraso e risco*. Se a Definição de Pronto não for igual a Potencialmente Entregável, então o trabalho será adiado até antes da liberação que causa o *risco e atraso*. Este trabalho atrasado às vezes é chamado *trabalho desfeito* (*undone work*).

Um Time Scrum (Scrum Team) deve melhorar continuamente, o que se reflete na extensão de sua Definição de Feito.

Planejamento de Sprints (Sprint Planning)

Resumo: Uma reunião de preparação para o Sprint, normalmente dividida em duas partes (a parte um é “o quê” e a parte dois é “como”).

Participantes: *Parte Um:* Dono do Produto, Equipe, Scrum Master.

Parte Dois: Equipe, Scrum Master, Dono do Produto (opcional, mas deve estar acessível em caso de dúvidas)

Duração: Cada parte tem um limite de tempo de uma hora por semana de Sprint.

No início de cada Sprint, a (Sprint Planning Meeting) **Reunião de Planejamento do Sprint** acontece em duas sub reuniões distintas, sendo a primeira denominada **Planejamento do Sprint Parte Um**.

Em **Planejamento do Sprint Parte Um**, o Product Owner e a Equipe revisam os itens de alta prioridade no Product Backlog que o Product Owner está interessado em implementar neste Sprint. Normalmente, esses itens terão sido bem analisados em um Sprint anterior (durante o Refinamento do Backlog do Produto), de modo que nesta reunião restem apenas pequenas dúvidas de esclarecimento de última hora. Nesta reunião, o Dono do Produto e a Equipe discutem os objetivos e o contexto desses itens de alta prioridade no Backlog do Produto, fornecendo à Equipe uma visão sobre o pensamento do Dono do Produto. A Parte Um se concentra na compreensão *do que* o Product Owner deseja e *por que* eles são necessários. No final da Parte Um, o Dono do Produto (sempre ocupado) pode sair, embora *deva* estar disponível (por exemplo, por telefone) durante a Parte Dois da reunião.

Na Parte Um, a Equipe e o Product Owner também podem elaborar o **Meta de sprint (Sprint Goal)**. Esta é uma declaração resumida do objetivo do Sprint, que idealmente tem um tema coeso. O Sprint Goal também dá à equipe flexibilidade de escopo em relação ao que eles podem realmente entregar, porque embora possam ter que remover algum item (já que o Sprint tem timeboxed), eles devem, no entanto, se comprometer a entregar algo tangível e “feito” que esteja no espírito do objetivo do Sprint.

Qual deve ser o tamanho dos itens assumidos em um Sprint? Cada item deve ser dividido em tamanho pequeno o suficiente para que seja estimado que exija consideravelmente menos do que todo o Sprint.

Uma diretriz comum é que um item seja estimado pequeno o suficiente para ser concluído dentro de um quarto ou menos de um Sprint por toda a equipe.

Planejamento do Sprint Parte Dois foca em *como* fazer para implementar os itens que a Equipe decide assumir. A equipe prevê a quantidade de itens que podem ser concluídos até o final do Sprint, começando no topo do Product Backlog (em outras palavras, começando com os itens que são de maior prioridade para o Product Owner) e descendo pela lista em ordem. *Esta é uma prática fundamental no Scrum: a equipe decide quanto trabalho será concluído, em vez de ser atribuído a eles pelo Dono do Produto.* Isto torna a previsão mais confiável porque a Equipe a faz com base em sua própria análise e planejamento. Embora o Product Owner não tenha controle sobre quanto a equipe inscreve, ele sabe que os itens são retirados do topo do Product Backlog – em outras palavras, os itens que ele ou ela classificou como mais importantes. A equipe tem a capacidade de fazer lobby por itens mais abaixo na lista; isso geralmente acontece quando a equipe e o Product Owner percebem que algo de menor prioridade se encaixa fácil e adequadamente nos itens de alta prioridade.

A Reunião de Planejamento do Sprint geralmente durará várias horas, mas não mais do que *quatro* horas para um Sprint de *duas* semanas – a equipe está fazendo uma previsão séria para concluir o trabalho, e isso requer uma reflexão cuidadosa para ter sucesso. A Parte Um e a Parte Dois têm durações de timebox iguais; para um Sprint de duas semanas, cada parte dura no máximo duas horas.

Scrum não define exatamente *como fazer* o Planejamento do Sprint Parte Dois. Algumas equipes usam a velocidade dos Sprints anteriores para orientar quanto almejar. Outras equipes usarão uma abordagem mais refinada, calculando primeiro sua capacidade.

Ao usar a abordagem de capacidade, a Equipe, na Segunda Parte do Planejamento do Sprint, calcula quanto tempo cada membro da equipe tem para o trabalho relacionado ao Sprint. A maioria das equipes presume que os membros da equipe só podem se concentrar no trabalho relacionado ao Sprint por 4 a 6 horas por dia – o resto do tempo é gasto em e-mails, intervalos para almoço, Facebook, reuniões e café. Uma vez determinada a capacidade, a equipe precisa descobrir quantos itens do Backlog do Produto eles podem concluir nesse período e como farão para concluí-los. Isso geralmente começa com uma discussão sobre design em um quadro branco. Uma vez compreendido o design geral, a equipe decompõe os itens do Product Backlog em um trabalho refinado. Antes de pegar os itens do Product Backlog, o Time pode focar na geração de tarefas para uma meta de melhoria criada na Retrospectiva da Sprint anterior. Em seguida, a equipe seleciona o primeiro item do Product Backlog – o item de maior prioridade do Product Owner – e desce até que estejam “cheios”. Para cada item eles criam uma lista de trabalho que consiste em itens decompostos do Backlog do Produto em tarefas ou, quando o item do Backlog do Produto é tão pequeno que levaria apenas algumas horas para ser implementado, simplesmente o item do Backlog do Produto. Esta lista de trabalhos a serem realizados durante o Sprint é chamada de **Pendências da Sprint(Sprint Backlog)** (Figura 4 e Figura 5).

				Novas estimativas de esforço					
				Restando no final do dia...					
Item do backlog do produto	Tarefa Sprint	Voluntário	Inicial Estimativa de Esforço	1	2	3	4	5	6
Como comprador, quero colocar um livro no carrinho de compras	modificar banco de dados		5						
	criar página da web (IU)		8						
	criar página da web (lógica Javascript)		13						
	escrever testes de aceitação automatizados		13						
	atualizar a página de ajuda do comprador		3						
	...								
	mesclar o código DCP e completar o nível da camada								

Melhore o desempenho do processamento de transações	testes		5							
	pedido completo de máquina para pRank		8							
	altere o DCP e o leitor para usar a API pRank http		13							

Figura 4. Exemplo de uma maneira de criar um Sprint Backlog

No final da Reunião de Planejamento do Sprint, a equipe define uma meta realista para o que acredita poder entregar até o final do Sprint. Tradicionalmente, isso era chamado de Compromisso do Sprint – a equipe se compromete a fazer o melhor que puder para atingir seu objetivo. Infelizmente, isso às vezes era mal interpretado como uma promessa escrita com sangue, em vez de a equipe seriamente “ir em frente”. Para evitar essa confusão, o alvo do sprint agora é chamado de “previsão” que é comunicada ao Dono do Produto.

Scrum incentiva trabalhadores multiquificados, em vez de apenas “trabalhar de acordo com o cargo”, como um “testador” apenas fazendo testes. Em outras palavras, os membros da Equipe “vão até onde está o trabalho” e ajudam na medida do possível. Se houver muitas tarefas de teste, então *todos* os membros da equipe podem ajudar. Isto não implica que todos sejam generalistas; sem dúvida, algumas pessoas são especialmente habilidosas em testes (e assim por diante), mas os membros da equipe trabalham juntos e aprendem novas habilidades uns com os outros. Consequentemente, durante a geração e estimativa de tarefas no Sprint Planning, não é necessário – nem apropriado – que as pessoas se voluntariem para todas as tarefas “que conseguem fazer melhor”. Em vez disso, é melhor voluntariar-se apenas para uma tarefa de cada vez, quando for altura de escolher uma nova tarefa, e considerar a escolha de tarefas que envolvam propositadamente aprendizagem (talvez através de trabalho em pares com um especialista). Esta é uma razão para não pré-atribuir tarefas durante o Planejamento do Sprint; em vez disso, isso deve ser feito “conforme necessário” durante o Sprint.

Dito tudo isso, existem *raros* momentos em que *John* pode realizar uma tarefa específica porque levaria muito tempo ou seria impossível para outros aprenderem – talvez John seja a única pessoa com alguma habilidade artística para fazer desenhos. Outros membros da Equipe não poderiam desenhar um “homem-palito” se suas vidas dependessem disso. Neste caso raro – e se não for raro e não ficar mais raro à medida que a Equipe aprende, há algo errado – pode ser necessário perguntar se o total de tarefas de desenho planejadas que *devem* ser feitas por John são viáveis dentro do Sprint curto.

Muitas equipes têm um Sprint Backlog na forma de um quadro de tarefas do tamanho de uma parede (geralmente chamado de **Quadro Scrum**) onde as tarefas (escritas em post-its) migram durante o Sprint através de colunas denominadas “A fazer” (To Do), “Trabalho em andamento”(Doing) e “Concluído” (Done). Veja a Figura 5.

Figura 5. Gerenciamento Visual – Tarefas do Sprint Backlog na parede



Um dos pilares do Scrum é que uma vez que o Time define sua meta para o Sprint, quaisquer adições ou alterações devem ser adiadas até o próximo Sprint. Isso significa que se no meio do Sprint o Product Owner decidir que há um novo item no qual ele gostaria que a equipe trabalhasse, ele não poderá fazer a alteração até o início do próximo Sprint. Se surgir uma circunstância externa que altere significativamente as prioridades e signifique que a Equipe estaria desperdiçando seu tempo se continuasse trabalhando, o Dono do Produto ou a Equipe podem encerrar o Sprint. A equipe para e faz uma nova reunião de planejamento do Sprint e inicia um novo Sprint. A perturbação de fazer isso geralmente é grande; isso serve como um desincentivo para o Dono do Produto ou Equipe recorrer a essa decisão dramática.

Há uma influência poderosa e positiva que vem do fato de a equipe estar protegida de mudanças de metas durante o Sprint. Primeiro, a Equipe começa a trabalhar sabendo com absoluta certeza que seu objetivo não mudará, o que reforça o foco da Equipe em garantir a conclusão. Em segundo lugar, disciplina o Product Owner a realmente pensar nos itens que ele prioriza no Product Backlog e oferece à equipe para o Sprint.

Ao seguir essas regras do Scrum, o Product Owner ganha duas coisas. Primeiro, ele ou ela tem a confiança de saber que a Equipe se comprometeu a fazer o melhor para concluir um conjunto de trabalho realista e claro que escolheu. Com o tempo, uma equipe pode se tornar bastante hábil na escolha e na entrega de previsões realistas. Em segundo lugar, o Dono do Produto pode fazer quaisquer alterações que desejar no Backlog do Produto antes do início do processo da *próxima* Sprint. Nesse ponto, adições, exclusões, modificações e redefinições de prioridade são todas possíveis e aceitáveis. Embora o Product Owner não seja capaz de fazer alterações nos itens selecionados em desenvolvimento durante o Sprint atual, ele ou ela está a apenas um Sprint ou menos de fazer as alterações que desejar. O estigma em torno da mudança acabou – mudança de direção, mudança de requisitos ou simplesmente mudança de idéia – e pode ser por esse motivo que os Proprietários do Produto geralmente ficam tão entusiasmados com o Scrum quanto qualquer outro.

Scrum Diário (Daily Scrum)

Resumo: Atualização e coordenação entre os membros da Equipe.

Participantes: Equipe é necessária; O Proprietário do Produto é opcional; O Scrum Master geralmente está presente, mas se não estiver deve assegurar que o time tenha a daily.

Duração: Duração máxima de 15 minutos.

Uma vez iniciado o Sprint, o Time se envolve em outra das principais práticas do Scrum: O **Scrum Diário(Daily)**. Esta é uma reunião curta (15 minutos ou menos) que acontece todos os dias úteis em um horário determinado. Todos da equipe comparecem. Para ser breve, recomenda-se que todos permaneçam em pé. É a oportunidade da Equipe sincronizar o seu trabalho e reportar uns aos outros sobre os obstáculos. No Daily Scrum, um por um, cada membro do Time relata três coisas *aos demais membros da equipe*: (1) O que foi realizado desde a última reunião?; (2) O que será feito antes da próxima reunião?; e (3) Que obstáculos estão no caminho? Observe que o Daily Scrum não é uma reunião de status para reportar a um gerente; é o momento para uma equipe auto-organizada compartilhar entre si o que está acontecendo, para ajudá-los a se coordenar. Alguém anota os bloqueios e o Scrum Master é responsável por ajudar os membros do Time a resolvê-los. Há pouca ou nenhuma discussão aprofundada durante o Daily Scrum, o tema é *fornecer* respostas às três perguntas; se a discussão for necessária, ela ocorre imediatamente após o Daily Scrum em uma ou mais reuniões paralelas de acompanhamento, embora no Scrum ninguém seja obrigado a comparecer a elas. Uma reunião de acompanhamento é um evento comum onde alguns ou todos os membros da equipe se adaptam às informações que ouviram no Daily Scrum: em outras palavras, outro ciclo de inspeção e adaptação. Para equipes novas no Scrum, geralmente é recomendado *não* ter gerentes ou outras pessoas em posições de autoridade percebida participando do Daily Scrum. Isto corre o risco de fazer com que a Equipe se sinta “monitorizada” – sob pressão para reportar grandes progressos todos os dias (uma expectativa irrealista) e inibida em reportar problemas – e tende a minar a autogestão da Equipe e a convidar à microgestão. Seria mais útil para uma parte interessada contactar a Equipe após a reunião e oferecer-se para ajudar com quaisquer bloqueios que estejam a atrasar o progresso da Equipe.

Acompanhando o progresso durante o Sprint

A Equipe no Scrum é auto gerenciada e, para fazer isso com sucesso, deve saber como está se saindo. Todos os dias, os membros da Equipe atualizam a sua estimativa do esforço restante para completar o seu trabalho atual no **Pendências da Sprint(Sprint Backlog)** (Figura 6). Também é comum alguém somar o esforço restante da Equipe como um todo e plotá-lo no **Gráfico Burndown do Sprint** (Figura 7 e Figura 8). Este gráfico mostra, a cada dia, uma nova estimativa de quanto trabalho resta até que a Equipe termine. Idealmente, este é um gráfico *descendente* inclinado que está em uma trajetória para atingir “zero esforço restante” no último dia do Sprint. Por isso é chamado de gráfico *burndown*. E embora às vezes pareça bom, muitas vezes não; esta é a realidade do desenvolvimento de produtos. O importante é que mostre à Equipe o progresso em direção ao objetivo, não em termos de quanto tempo foi *gasto* no passado (um fato irrelevante em termos de *progresso*), mas em termos de quanto trabalho *permanece no futuro* – o que separa a equipe de seu objetivo. Se a linha de burndown não estiver descendo até a conclusão perto do final do Sprint, então a equipe precisa se ajustar, como reduzir o escopo do trabalho ou encontrar uma maneira de trabalhar de forma mais eficaz, mantendo um ritmo sustentável.

Embora o gráfico Sprint Burndown possa ser criado e exibido usando uma planilha, muitas equipes acham mais eficaz mostrá-lo em papel na parede de seu espaço de trabalho, com atualizações em caneta; esta solução de “baixa tecnologia/alto contato” é rápida, simples e muitas vezes mais visível do que um gráfico de computador.

				Novas estimativas de esforço					
				Restando no final do dia...					
Item do backlog do produto	Tarefa Sprint	Voluntário	Inicial Estimativa de Esforço	1	2	3	4	5	6

Como comprador, quero colocar um livro em um carrinho de compras	modificar banco de dados	Sanjay	5	4	3	0	0	0	
	criar página da web (IU)	Jing	3	3	3	2	0	0	
	criar página da web (lógica Javascript)	Tracy e Sam	2	2	2	2	1	0	
	escrever testes de aceitação automatizados	Sara	5	5	5	5	5	0	
	atualizar a página de ajuda do comprador	Sanjay e Jing	3	3	3	3	3	0	
	...								
Melhore o desempenho do processamento de transações	mesclar código DCP e concluir testes em nível de camada		5	5	5	5	5	5	
	pedido completo de máquina para pRank		3	3	8	8	8	8	
	altere o DCP e o leitor para usar a API pRank http		5	5	5	5	5	5	
...			...						
			Total	50	49	48	44	43	34

Figura 6. Atualizações diárias do trabalho restante no Sprint Backlog

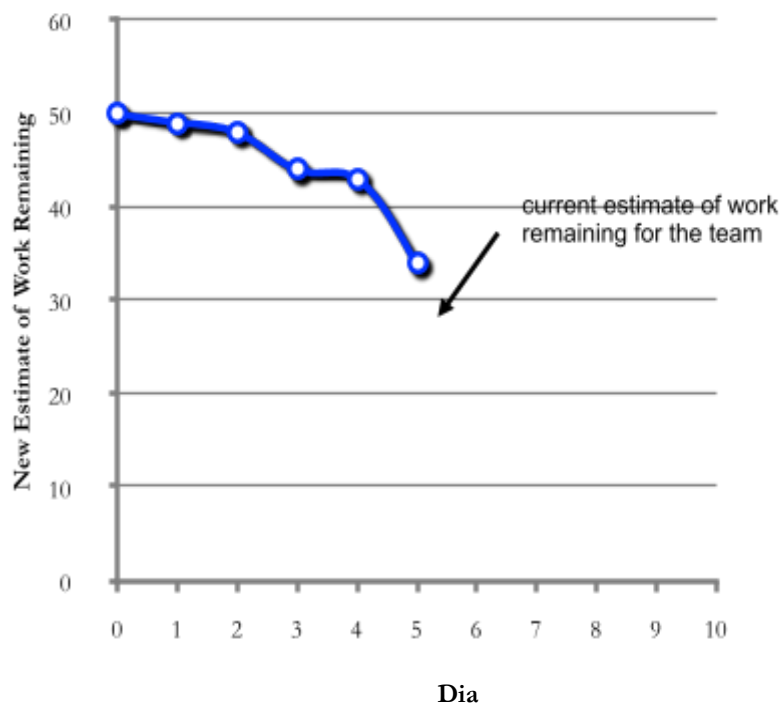


Figura 7. Gráfico Burndown do Sprint

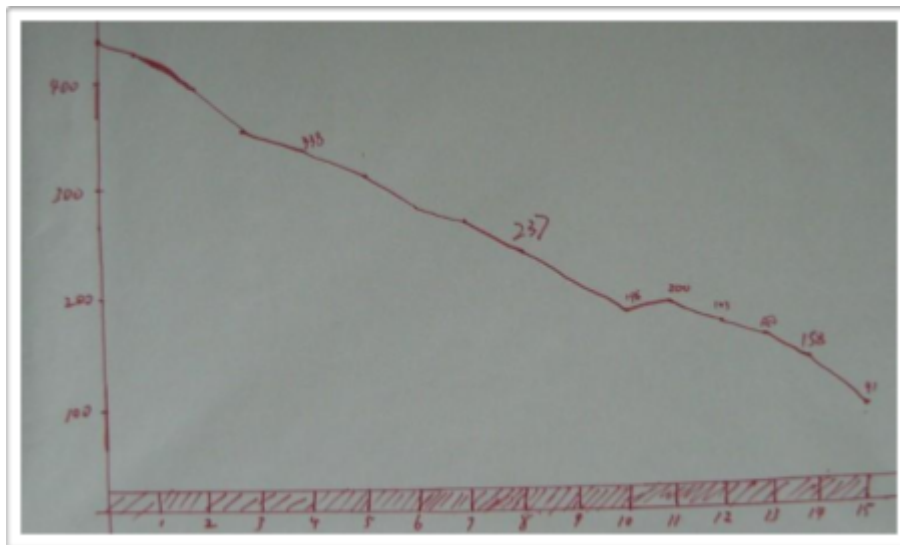


Figura 8. Gerenciamento Visual: Gráfico Burndown do Sprint Desenhado à Mão

Refinamento do Backlog do Produto

Resumo: Divida itens grandes, analise itens, reestime e repriorize, por Sprint futura.

Participantes: Equipe; O Product Owner participará de toda a atividade se for o especialista que pode ajudar no refinamento detalhado, caso contrário, poderá participar apenas de um subconjunto para definir a direção ou repriorizar; outros que entendam os requisitos e possam ajudar a Equipe; O Scrum Master comparecerá durante as sessões iniciais para treinar o grupo para ser eficaz, caso contrário poderá não comparecer.

Duração: Normalmente, não mais que 10% da capacidade do Time para o Sprint, embora possa ser maior para itens “pesados em análise”. Por exemplo, em um Sprint de duas semanas, talvez um dia seja gasto em refinamento.

Uma das diretrizes menos conhecidas, mas valiosas, do Scrum é que uma porcentagem de cada Sprint deve ser dedicada por toda a equipe para refinar (ou “preparar”) o Backlog do Produto para apoiar Sprints futuros. Isso inclui análise detalhada de requisitos, divisão de itens grandes em itens menores, estimativa de novos itens e reestimação de itens existentes. O Scrum não fala sobre como esse trabalho é feito, mas uma técnica frequentemente utilizada é um workshop focado próximo ao meio ou final do Sprint, para que o Time e o Product Owner e demais stakeholders possam se dedicar a esse trabalho sem interrupção.

Esta atividade de refinamento é para itens *não* selecionados para o Sprint atual; é para itens para o futuro, provavelmente nos próximos um ou dois Sprints. Com esta prática, o Planejamento do Sprint se torna relativamente simples porque o Product Owner e o Time Scrum iniciam o planejamento com um conjunto de itens claros, bem analisados e cuidadosamente estimulados. Um sinal de que esse workshop de refinamento não está sendo realizado (ou não está sendo bem realizado) é que o Planejamento do Sprint envolve questões significativas, descobertas ou confusão e parece incompleto; o trabalho de planejamento geralmente se estende ao próprio Sprint, o que normalmente não é desejável.

Revisão da Sprint(Review)

Resumo: Inspeção e adequação relacionadas ao incremento de funcionalidade do produto.

Participantes: Equipe, Dono do Produto, Scrum Master. Outras partes interessadas, conforme apropriado, convidadas pelo Product Owner.

Duração: Timebox para uma hora por semana de Sprint.

Após o término do Sprint, há o **Revisão da Sprint (Sprint Review)**, onde as pessoas revisam o Sprint. Estão presentes nesta reunião o Product Owner, os membros da equipe e o Scrum Master, além de clientes, usuários, stakeholders, especialistas, executivos e qualquer pessoa interessada. Para um Sprint de duas semanas a duração máxima é de duas horas. Qualquer pessoa presente é livre para fazer perguntas e dar sugestões.

A Revisão é frequentemente rotulada erroneamente como “demonstração”, mas isso não capta a real intenção desta reunião. Uma ideia chave no Scrum é *inspecionar e adaptar*. Ver e aprender o que está acontecendo e depois evoluir com base no feedback, na repetição de ciclos. A Sprint Review é uma atividade de inspeção e adaptação para os *produtos*. É um momento para o Product Owner saber o que está acontecendo com o produto e com o Time (ou seja, uma revisão da Sprint); e para a Equipe saber o que está acontecendo com o Product Owner e o mercado. Consequentemente, um elemento crítico da Revisão é uma análise aprofundada da *conversação* entre a equipe e o proprietário do produto para conhecer a situação, obter conselhos e assim por diante. A revisão definitivamente inclui o uso do *software real* que a equipe construiu durante o Sprint, mas se o foco da revisão for apenas olhar para o produto em vez de conversar, há um desequilíbrio.

A parte do “software ao vivo” da Sprint Review não é uma “apresentação” feita pela equipe – *não há slideware*. O objetivo é ser uma inspeção prática do software real em execução, por exemplo, em um ambiente de desenvolvimento sandbox. Haverá um ou mais computadores na sala de revisão onde as pessoas poderão inspecionar e usar o software ativo. Prefira uma sessão ativa na qual usuários reais e o Product Owner interajam de forma prática com o software, em vez de uma sessão de demonstração passiva da equipe.

Procure não gastar mais do que 30 minutos se preparando para a Sprint Review, caso contrário, isso sugere que algo está errado.

Retrospectiva da Sprint (Sprint Retrospective)

Resumo: Inspeção e adequação relacionadas ao processo e ambiente.

Participantes: Equipe, Scrum Master, Proprietário do produto (opcional). Outras partes interessadas poderão ser convidadas pela Equipe, mas não poderão participar de outra forma.

Duração: Timebox definido para 45 minutos por semana de Sprint.

A Sprint Review envolve inspecionar e adaptar em relação aos *produtos*. O **Retrospectiva da Sprint (Sprint Review)**, que segue a Revisão, envolve inspecionar e adaptar em relação ao *processo e ambiente*. É uma oportunidade para a equipe discutir o que funciona e o que não funciona e chegar a um acordo sobre as mudanças a serem testadas. Às vezes, o Scrum Master pode atuar como um facilitador eficaz para a Retrospectiva, mas pode ser melhor encontrar uma pessoa externa neutra para facilitar a reunião; uma boa abordagem é os Scrum Masters facilitarem as retrospectivas uns dos outros, o que permite a polinização cruzada entre as equipes.

Existem muitas técnicas para conduzir uma Retrospectiva Sprint, e o livro *Retrospectivas Ágeis* (Derby, Larsen 2006) fornece um catálogo útil de técnicas.

Muitas equipes realizam retrospectivas com foco apenas em *problemas*, e isso é uma pena. Isso pode levar as pessoas a pensarem nas retrospectivas como eventos um tanto deprimentes ou negativos. Em vez disso, certifique-se de que cada Retrospectiva também se concentra nos pontos positivos e fortes; existem vários livros sobre *investigação Apreciativa* que oferecem dicas mais detalhadas.

Retrospectivas que utilizam sempre a mesma técnica de análise podem tornar-se enfadonhas; portanto, introduza várias técnicas ao longo do tempo.

Iniciando o próximo Sprint

Após a Revisão do Sprint, o Dono do Produto pode atualizar o Backlog do Produto com qualquer novo insight – *adicionando* novos itens, *removendo* itens obsoletos ou *revisando* itens existentes. O Dono do

Produto é responsável por garantir que essas mudanças sejam refletidas no Backlog do Produto. Veja a Figura 9 para obter um exemplo do Backlog do Produto atualizado.

				Novas estimativas na Sprint ...				
Prioridade	Item	Detalhes (wiki URL)	Inicial Estimativa	1	2	3	4	5
1	Como comprador, quero colocar um livro em um carrinho de compras (veja os esboços da IU na página wiki)	...	5	0	0	0		
2	Como comprador, quero remover um livro do carrinho de compras	...	2	0	0	0		
3	Melhore o desempenho do processamento de transações (veja as métricas de desempenho alvo na wiki)	...	13	13	0	0		
4	Investigue soluções para acelerar a validação de cartão de crédito (veja as métricas de desempenho alvo na wiki)	...	20	20	20	0		
5	Atualize todos os servidores para Apache 2.2.3	...	13	13	13	13		
6	Diagnosticar e corrigir erros de script de processamento de pedidos (ID do bugzilla 14823)	...	3	3	3	3		
7	Como comprador, quero criar e salvar uma lista de desejos	...	40	40	40	40		
8	Como comprador, quero adicionar ou excluir itens da minha lista de desejos	...	20	20	20	20		
...								
			537	580	570	500		

Figura 9. Backlog do produto atualizado

Não há tempo de inatividade entre os Sprints – As equipes normalmente passam de uma Retrospectiva do Sprint em uma tarde para o Planejamento do Sprint seguinte na manhã seguinte (ou após o fim de semana).

Um dos princípios do desenvolvimento ágil é o “ritmo sustentável”, e somente trabalhando horas regulares em um nível razoável as equipes podem continuar este ciclo indefinidamente. A produtividade cresce ao longo do tempo através da evolução das práticas da Equipe e da remoção de impedimentos à produtividade da Equipe, e não através do excesso de trabalho ou do comprometimento da qualidade.

Os sprints continuam até que o Product Owner decida que o produto está pronto para lançamento. A visão de perfeição do Scrum é que o produto seja potencialmente entregue no final de cada Sprint, o que implica que não há necessidade de trabalho de conclusão, como testes ou documentação. A implicação é que *tudo* é completamente *finalizado* a cada Sprint; que você poderia realmente enviá-lo ou implantá-lo imediatamente após a Sprint Review. No entanto, muitas organizações têm práticas de desenvolvimento, ferramentas e infra-estruturas fracas e não conseguem alcançar esta visão de perfeição e por isso haverá a necessidade de um “Release Sprint” para lidar com este trabalho restante. Quando um “Release Sprint” é necessário, é considerado um mal necessário e o trabalho da organização é melhorar suas práticas para que isso não seja mais necessário.

Gerenciando versões

Uma pergunta que às vezes é feita é como, em um modelo iterativo, pode ser feito o planejamento de lançamentos de longo prazo. Há dois casos a serem considerados: (1) um produto novo em sua primeira versão e (2) um produto existente em uma versão posterior.

No caso de um novo produto ou *um produto existente que acaba de adotar o Scrum*, há a necessidade de fazer o refinamento inicial do Backlog do Produto antes do primeiro Sprint, onde o Dono do Produto e a Equipe moldam um Backlog do Produto Scrum adequado. Isso pode levar alguns dias ou uma semana e envolve um workshop (às vezes chamado de Criação Inicial do Backlog do Produto ou Planejamento de Release), algumas análises detalhadas de requisitos e estimativa de todos os itens identificados para o primeiro lançamento.

Surpreendentemente, no Scrum, no caso de um produto estabelecido com um Backlog de Produto estabelecido, não deve haver necessidade de nenhum planejamento de lançamento especial ou extenso para o próximo lançamento. Por que? Porque o Product Owner e a Equipe devem fazer o refinamento do Product Backlog a cada Sprint (cinco ou dez por cento de cada Sprint), preparando-se continuamente para o futuro. Esse *desenvolvimento contínuo de produtos* evita a necessidade dos dramáticos estágios pontuados de preparar-executar-concluir que vemos no desenvolvimento sequencial tradicional do ciclo de vida.

Durante um workshop inicial de refinamento do Backlog do Produto e durante o refinamento contínuo do backlog a cada Sprint, a Equipe e o Dono do Produto farão o planejamento da versão, refinando as estimativas, prioridades e conteúdo à medida que aprendem.

Alguns lançamentos são baseados em datas; por exemplo: “Lançaremos a versão 2.0 do nosso projeto em uma feira no dia 10 de novembro.” Nesta situação, o Time completará tantos Sprints (e construirá tantos recursos) quanto possível no tempo disponível. Outros produtos exigem que certos recursos sejam construídos antes que possam ser considerados completos e o produto não será lançado até que esses requisitos sejam atendidos, por mais que demore. Como o Scrum enfatiza a produção de código potencialmente distribuível a cada Sprint, o Product Owner pode optar por começar a fazer lançamentos provisórios, para permitir que o cliente colha os benefícios do trabalho concluído mais cedo.

Como eles não podem saber tudo antecipadamente, o foco está na criação e no refinamento de um plano para dar uma direção ampla ao lançamento e esclarecer como as decisões de compensação serão tomadas (escopo versus cronograma, por exemplo). Pense nisso como um roteiro que o guiará até o seu destino final; quais estradas exatas você toma e as decisões que você toma durante a viagem podem ser determinadas no caminho.

O destino é mais importante que a viagem.

A maioria dos Product Owners escolhe uma abordagem de lançamento. Por exemplo, eles decidirão uma data de lançamento e trabalharão com a equipe para estimar os itens do Backlog do Produto que podem ser concluídos até essa data. Os itens previstos para a versão atual são às vezes chamados de *liberar itens*. Em situações em que é necessário um compromisso de “preço fixo/data fixa/entregável fixo” – por exemplo, desenvolvimento de contrato – um ou mais desses parâmetros devem ter um buffer integrado para permitir incertezas e mudanças; nesse aspecto, o Scrum não é diferente de outras abordagens.

Foco na aplicação ou produto

Para aplicações ou produtos – seja para o mercado ou para uso interno dentro de uma organização – o Scrum afasta os grupos do antigo modelo centrado em projetos em direção a um *desenvolvimento contínuo de aplicações/produtos* modelo. Não existe mais um projeto com começo, meio e fim. E, portanto, nenhum gerente de projeto tradicional. Em vez disso, existe simplesmente um Product Owner estável e uma equipe autogerenciada de longa duração que colaboram em uma série “infinita” de Sprints de duração fixa, até que o produto ou aplicativo seja retirado. Todo o trabalho necessário de gerenciamento de “projetos” é feito pela Equipe e pelo Dono do Produto – que é um cliente comercial interno ou do Gerenciamento de Produto. Não é gerenciado por um gerente de TI ou alguém de um escritório de gerenciamento de projetos.

Scrum também pode ser usado para verdadeiros *projetos* que são iniciativas únicas (em vez de trabalho para criar ou desenvolver aplicações de longa duração); ainda assim, neste caso a Equipe e o Product Owner fazem o gerenciamento do projeto.

E se não houver trabalho novo suficiente em um ou mais aplicativos existentes para garantir uma equipe dedicada e duradoura para cada aplicativo? Nesse caso, uma equipe estável e de longa duração pode assumir itens de um aplicativo em um Sprint e, em seguida, itens de outro no próximo Sprint; nesta situação, os Sprints costumam ser bastante curtos, como uma semana.

Ocasionalmente, não há trabalho novo suficiente, mesmo para a solução anterior, e a Equipe pode assumir itens de *diversas* aplicações durante o mesmo Sprint; no entanto, tome cuidado com esta solução, pois ela pode evoluir para multitarefa improdutiva em vários aplicativos. Um tema básico de produtividade no Scrum é que a equipe esteja *focada* em um produto ou aplicativo para um Sprint.

Desafios Comuns

Scrum não é apenas um conjunto concreto de práticas – pelo contrário, e mais importante, é uma estrutura que proporciona transparência e um mecanismo que permite “inspecionar e adaptar”. O Scrum funciona tornando visíveis as disfunções e os impedimentos que estão impactando o Product Owner e a eficácia do Time, para que possam ser resolvidos. Por exemplo, o Product Owner pode não conhecer realmente o mercado, os recursos ou como estimar o valor relativo do negócio. Ou a equipe pode ser inábil na estimativa de esforço ou no trabalho de desenvolvimento.

A estrutura Scrum revelará rapidamente essas fraquezas. Scrum não resolve os problemas de desenvolvimento; torna-os dolorosamente visíveis e fornece uma estrutura para as pessoas explorarem formas de resolver problemas em ciclos curtos e com pequenas experiências de melhoria.

Suponha que a equipe não consiga entregar o que prevê no primeiro Sprint devido à falta de análise de tarefas e habilidade de estimativa. Para a equipe, isso parece um fracasso. Mas, na realidade, esta experiência é o primeiro passo necessário para se tornar mais realista e ponderado sobre as suas previsões. Este padrão – de Scrum ajudando a tornar a disfunção visível, permitindo que a equipe faça algo a respeito – é o mecanismo básico que produz os benefícios mais significativos que as equipes que usam Scrum experimentam.

Um erro comum cometido, quando apresentado a uma prática Scrum desafiadora, é mudar o Scrum. Por exemplo, as equipes que têm problemas para entregar podem decidir tornar a duração do Sprint extensível, para que o tempo nunca acabe – e no processo, garantir que nunca precisem aprender como fazer um trabalho melhor de estimativa e gerenciamento de seu tempo. Desta forma, sem coaching e o apoio de um Scrum Master experiente, as organizações podem transformar o Scrum em apenas uma imagem espelhada de suas próprias fraquezas e disfunções, e minar o benefício real que o Scrum oferece: Tornar visíveis o que é bom e o que é ruim, e dar a organização a escolha de se elevar a um nível superior.

Outro erro comum é presumir que uma prática é desencorajada ou proibida apenas porque o Scrum não a exige especificamente. Por exemplo, Scrum não exige que o Product Owner estabeleça uma estratégia de longo prazo para seu produto; nem exige que os engenheiros procurem aconselhamento de engenheiros mais experientes sobre problemas técnicos complexos. Scrum deixa para os indivíduos envolvidos tomarem a decisão certa; e na maioria dos casos, ambas as práticas (juntamente com muitas outras) são bem aconselhadas.

Outra coisa a ter em conta são os gestores que impõem o Scrum às suas equipes; Scrum é dar espaço e ferramentas à equipe para gerenciar a si mesma, e ter isso ditado de cima não é uma receita para o sucesso. Uma abordagem melhor pode começar com uma equipe aprendendo sobre Scrum com um colega ou gerente, recebendo formação abrangente em treinamento profissional e, em seguida, tomando uma decisão como equipe de seguir as práticas fielmente por um período definido; ao final desse período, a Equipe avaliará sua experiência e decidirá se deseja continuar.

A boa notícia é que embora o primeiro Sprint seja geralmente muito desafiador para o Time, os benefícios do Scrum tendem a ser visíveis no final dele, levando muitos novos Times Scrum a exclamar: “Scrum é difícil, mas com certeza é um todo muito melhor do que estávamos fazendo antes!”

Apêndice A: Leitura Adicional

Há muito material publicado sobre Scrum. Nesta seção de referência, gostaríamos de destacar algum material online adicional e alguns livros.

Materiais on-line:

- [The Lean Primer - Uma introdução ao Lean Thinking, uma influência importante para o Scrum. http://www.leanprimer.com](http://www.leanprimer.com)
- [The Distributed Scrum Primer – Dicas adicionais para equipes que não estão co-localizadas. http://www.goodagile.com/distributedscrumprimer/](http://www.goodagile.com/distributedscrumprimer/)
- [The Scrum Master Checklist – Uma lista de perguntas que bons Scrum Masters usam. http://www.ScrumMasterchecklist.org/](http://www.ScrumMasterchecklist.org/)
- [Feature Team Primer - Dimensionando o Scrum com Feature Teams. http://www.featureteams.org](http://www.featureteams.org)
- [O Atlas Ágil - Core Scrum. Descrição ScrumAlliance do Scrum. http://agileatlas.org/atlas/scrum](http://agileatlas.org/atlas/scrum)
- [Guia Scrum - descrição do Scrum.org em Scrum. http://www.scrum.org/Scrum-Guides](http://www.scrum.org/Scrum-Guides)
- [Agile Contracts Primer – Como fazer contratos amigáveis ao Scrum. http://www.agilecontracts.org/](http://www.agilecontracts.org/)

Livros:

- Leading Teams - Richard Hackman
- Scaling Lean & Agile Development: Thinking and Organizational Tools for Large-Scale Scrum - Craig Larman, Bas Vodde
- Practices for Scaling Lean & Agile Development: Large, Multisite, and Offshore Product Development with Large-Scale Scrum - Craig Larman, Bas Vodde
- Agile Project Management with Scrum - Ken Schwaber
- Succeeding with Agile: Software Development using Scrum - Mike Cohn

Apêndice B: Terminologia

Queimar

A tendência do trabalho restante ao longo do tempo em um Sprint, Release ou Produto. A fonte dos dados brutos é o Sprint Backlog e o Product Backlog, com o trabalho restante rastreado no eixo vertical e os períodos de tempo (dias de um Sprint ou Sprints) rastreados no eixo horizontal.

Scrum Diário

Uma breve reunião realizada diariamente por cada Equipe durante a qual os membros da Equipe inspecionam seu trabalho, sincronizam seu trabalho e progresso e relatam impedimentos ao Scrum Master para remoção. Reuniões de acompanhamento para adaptar o trabalho futuro para otimizar o Sprint podem ocorrer após as reuniões do Daily Scrum.

Equipe de desenvolvimento

Outro nome para a função de Equipe.

Feito

Preencher conforme acordado mutuamente por todas as partes e que esteja em conformidade com os padrões, convenções e diretrizes da organização. Quando algo é relatado como “feito” na reunião de Revisão do Sprint, deve estar de acordo com esta definição acordada.

Trabalho restante estimado (itens do Backlog do Sprint)

O número de horas que um membro da equipe estima que restam para trabalhar em qualquer tarefa. Essa estimativa é atualizada ao final de cada dia quando a tarefa do Sprint Backlog é trabalhada. A estimativa é o esforço total estimado restante, independentemente do número de pessoas que realizam o trabalho.

Incremento

Funcionalidade do produto desenvolvida pela equipe durante cada Sprint que pode ser potencialmente entregue ou usada pelas partes interessadas do Dono do Produto.

Incremento da Funcionalidade do Produto Potencialmente Entregável

Uma fatia completa do produto ou sistema geral que poderia ser usada pelo Dono do Produto ou pelas partes interessadas se eles decidissem implementá-lo.

Sprint

Uma iteração, ou um ciclo repetido de trabalho semelhante, que produz um incremento no produto ou sistema. Não mais que um mês e geralmente mais que uma semana. A duração é fixa ao longo de todo o trabalho e todas as equipes que trabalham no mesmo sistema ou produto utilizam o mesmo ciclo de duração.

Backlog do produto

Uma lista priorizada de requisitos com tempos estimados para transformá-los em funcionalidades completas do produto. As estimativas são mais precisas quanto mais alto um item estiver na prioridade do Backlog do Produto. A lista emerge, mudando conforme as condições de negócios ou a tecnologia mudam.

Item do backlog do produto

Requisitos funcionais, requisitos não funcionais e problemas, priorizados em ordem de importância para o negócio e dependências, e estimados. A precisão da estimativa depende da prioridade e granularidade do item do Product Backlog, sendo os itens de maior prioridade que poderão ser selecionados no próximo Sprint muito granulares e precisos.

Proprietário do produto

A pessoa responsável por gerenciar o Product Backlog de forma a maximizar o valor do produto. O Product Owner é responsável por representar os interesses de todos os envolvidos no projeto e no produto resultante.

Scrum

Não é um acrônimo, mas mecanismos no jogo de rugby para colocar uma bola fora de jogo de volta em jogo.

Scrum Master

O responsável pelo processo Scrum, sua correta implementação e maximização de seus benefícios.

Pendências da Sprint

Uma lista do trabalho da equipe para um Sprint. Muitas vezes, isso é decomposto em um conjunto de tarefas mais detalhadas. A lista surge durante o Planejamento do Sprint e pode ser atualizada pela equipe durante o Sprint com itens sendo removidos ou novas tarefas sendo adicionadas conforme

necessário. Cada tarefa do Sprint Backlog será rastreada durante o Sprint e mostrará o esforço restante estimado.

Tarefa do Backlog do Sprint

Uma das tarefas que a Equipe ou um membro da Equipe define como necessária para transformar itens comprometidos do Backlog do Produto em funcionalidade do sistema.

Reunião de planejamento da sprint

Um tempo de reunião limitado a quatro horas (para um Sprint de duas semanas) que inicia cada Sprint. A reunião é dividida em dois segmentos de duas horas, cada um também delimitado por tempo. Durante a primeira parte, o Product Owner apresenta o Product Backlog de maior prioridade para a equipe. A Equipe e o Product Owner colaboram para ajudar a Equipe a determinar quanto o Product Backlog pode transformar em funcionalidade durante o próximo Sprint. Durante a segunda parte, a equipe planeja como conseguirá isso, projetando e decompondo o trabalho para que entendam como atingirão a meta do Sprint.

Reunião de retrospectiva da Sprint

Uma reunião facilitada pelo Scrum Master na qual todo o Time discute o Sprint recém-concluído e determina o que poderia ser mudado para tornar o próximo Sprint mais agradável ou produtivo.

Reunião de revisão do sprint

Uma reunião com limite de tempo de duas horas (para um Sprint de duas semanas) no final de cada Sprint, onde a equipe colabora com o Dono do Produto e as partes interessadas e inspecionam o resultado do Sprint. Isso geralmente começa com uma revisão dos itens concluídos do Backlog do Produto, uma discussão sobre oportunidades, restrições e riscos, e uma discussão sobre quais podem ser as melhores coisas a fazer a seguir (resultando potencialmente em alterações no Backlog do Produto). Apenas a funcionalidade completa do produto pode ser demonstrada.

Parte interessada

Alguém com interesse no resultado de um projeto, seja porque o financiou, irá utilizá-lo ou será afetado por ele.

Equipe

Um grupo multifuncional de pessoas responsável por gerenciar a si mesmos para desenvolver um incremento de produto a cada Sprint.

Timebox

Um período de tempo que não pode ser excedido e dentro do qual ocorre um evento ou reunião. Por exemplo, uma reunião Daily Scrum tem o tempo fixado em quinze minutos e termina ao final de quinze minutos, independentemente. Para reuniões, pode durar menos. Para Sprints, dura exatamente essa duração.