

AGILE & MANAGEMENT 3.0

SCRUM NA PRÁTICA

ANDRÉ LUIZ DIAS RIBEIRO



6

LISTA DE FIGURAS

Figura 6.1 – Framework Scrum	5
Figura 6.2 – Analogia a Sprint	6
Figura 6.3 – Planejamento Sprint	8
Figura 6.4 – Analogia a reunião	8
Figura 6.5 – Fluxo do Planejamento do Sprint	10
Figura 6.6 – Product Backlog no quadro Kanban	11
Figura 6.7 - Checklist.....	12
Figura 6.8 - Processos concluídos	14
Figura 6.9 - PO.....	16
Figura 6.10 – Execução do Sprint	17
Figura 6.11 – Quadro Kanban durante a execução de um Sprint	18
Figura 6.12 – Reunião diária	19
Figura 6.13 – Product Burndown.....	20
Figura 6.14 – Sprint Burndown.....	21
Figura 6.15 – Review Meeting.....	22
Figura 6.16 – Retrospectiva	24

LISTA DE QUADROS

Quadro 6.1 – Timebox dos Sprint Planning.....	7
---	---

EXEMPLO

SUMÁRIO

6 SCRUM NA PRÁTICA.....	5
6.1 Planejamento da Sprint	5
6.2 Etapas do planejamento do Sprint	8
6.3 Sprint Backlog	10
6.3.1 Conceito de Ready.....	11
6.3.2 Reunião de Grooming	13
6.3.3 Conceito de PRONTO	13
6.4 Testes de aceitação	14
6.5 Execução do Sprint	17
6.5.1 Um Sprint pode ser cancelado?	18
6.5.2 Reunião diária	19
6.5.3 Burndown Chart.....	20
6.6 Revisão do Sprint	22
6.6.1 Resultados da revisão	23
6.6.2 Retrospectiva	23
6.7 Dicas	24
REFERÊNCIAS	27

6 SCRUM NA PRÁTICA

6.1 Planejamento da Sprint

Dentro do ciclo do *framework Scrum*, após a definição do Product Backlog, inicia-se a reunião de planejamento que é dividida em duas etapas: o planejamento do Product Backlog e planejamento do Sprint, como mostra a figura a seguir:

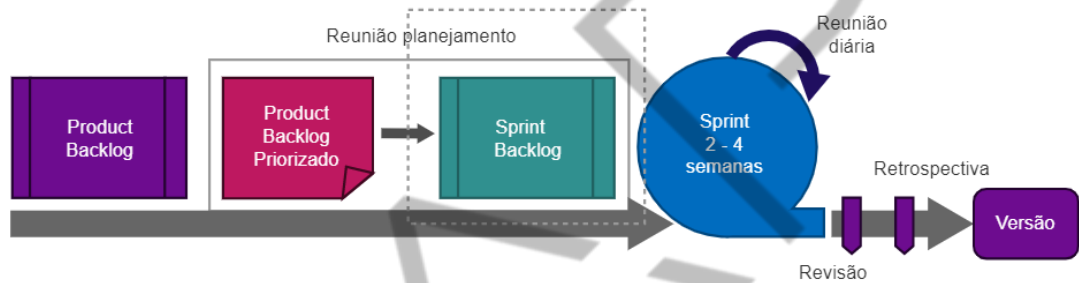


Figura 6.1 – Framework Scrum
Fonte: Elaborado pelo autor, adaptado por FIAP (2017)

Vimos, no capítulo anterior, a primeira etapa desse ciclo. Agora, veremos em detalhes como fazer o planejamento do Sprint, que será executado. Mas antes disso, vamos entender o que é um Sprint. É um ciclo *timebox* para construir o produto definido. Dentro do Sprint podemos usar o método que quisermos para a produção, desde que o padrão de qualidade seja observado e o prazo seja respeitado. Lembre-se de que as histórias que compõem um Sprint são definidas pelo Product Owner e suas atividades e compromisso com o prazo são de responsabilidade do time.

No Scrum, cada Sprint é um timebox, que dura de duas a quatro semanas e sempre deve entregar valor ao cliente. A duração de cada Sprint é definida para cada projeto, em comum acordo entre todos os envolvidos, e pode variar de duração de acordo com as necessidades.

IMPORTANTE: Lembre-se de que o conceito de *timebox* é que a data final é definitiva, não pode ser prorrogada nem antecipada sob nenhuma hipótese.

Não existe tamanho recomendado para um Sprint, isso vai depender da quantidade de histórias, do time e do que espera o Product Owner como meta do

Sprint. Eles também podem ter tamanhos variados, dentro de um projeto, e esse tamanho varia de acordo com o que for definido entre o cliente e o time.

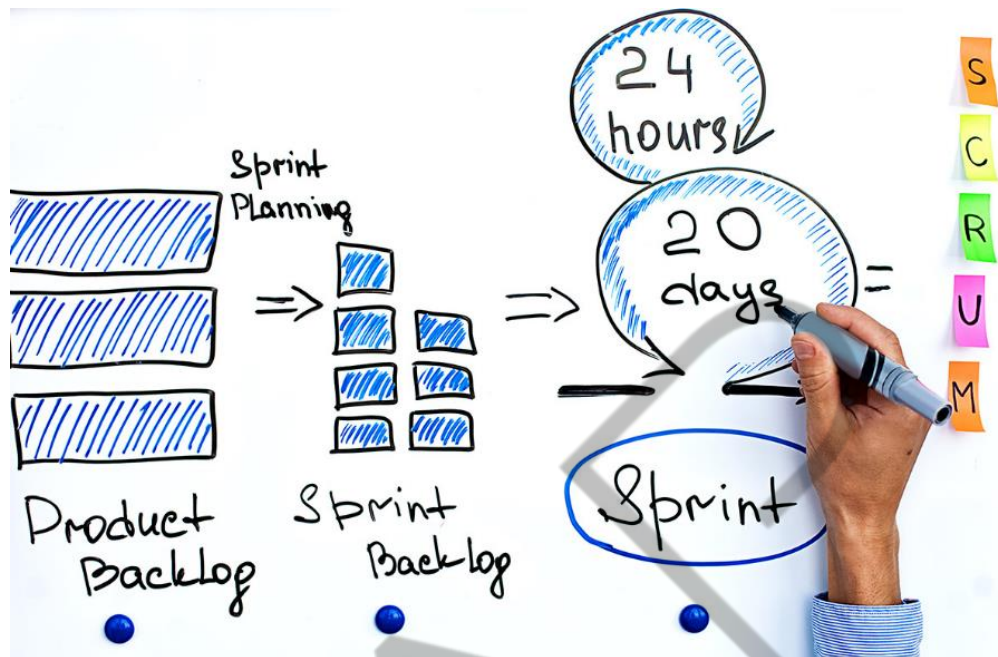


Figura 6.2 – Analogia a Sprint
Fonte: Banco de imagens Shutterstock (2017)

Embora, não existam padrões estabelecidos para o Sprint, podemos ter como orientação inicial para definir o tamanho:

- Time começando a executar projetos com Scrum deve usar Sprint de duas semanas para que as adaptações sejam mais rápidas.
- Primeiros Sprints de um projeto novo devem ser de duas semanas para minimizar os riscos e adaptar o time ao ambiente do projeto.
- Product Backlog tem muitas mudanças de priorização pelo Product Owner, que pode indicar indecisões do Product Owner, usar Sprint de duas semanas.
- Use Sprints de três ou quatro semanas para reduzir o ritmo ou aliviar o esforço do time, durante o projeto ou no Sprint anterior à liberação de uma versão.

Além dos requisitos de negócio, um Sprint pode ter atividades técnicas que devem ser colocadas junto às necessidades de negócio para atender às expectativas e preocupações técnicas do time. Como, por exemplo: requisitos de desempenho, requisitos de usabilidade, requisitos de segurança, arquitetura, entre outras.

Antes de iniciar o Sprint, deve ser definida uma META específica que esteja alinhada à meta do release e atenda às expectativas do cliente. O Product Owner é responsável por definir essa meta. A meta é o guia do time, durante a execução do trabalho e deve ser atendida dentro do timebox que foi definido.

IMPORTANTE: O Scrum Master deve garantir que nenhuma mudança possa afetar a meta do Sprint.

O Sprint Planning é a segunda parte da reunião de planejamento e é realizada, antes da execução de cada Sprint, quando o time e o Product Owner se reúnem para detalhar o que será feito durante o timebox de construção. Ele também é uma reunião do tipo *timebox*, definido no Scrum Guide®, que deve ter seu tempo de duração fixo, conforme mostra o quadro: “Timebox dos Sprint Planning”.

Tamanho Sprint	Timebox da reunião
4 semanas	8 horas
3 semanas	6 horas
2 semanas	4 Horas

Quadro 6.1 – Timebox dos Sprint Planning
Fonte: Elaborado pelo autor (2017)

A duração desses eventos deve ser respeitada incondicionalmente por todos os envolvidos no projeto e seu cumprimento deve ser pelo Scrum Master.

Embora, o Scrum defina Sprints com tamanhos entre duas e quatro semanas, caso necessário, pode-se fazer um Sprint de uma semana para a execução de ciclos menores de atividades. Mas, não pode ser maior que quatro semanas.

Na reunião de planejamento do Sprint, deve ser definido o que precisa ser entregue e, em seguida, definir como fazer para entregar o trabalho. Vamos analisar, em etapas.



Figura 6.3 – Planejamento Sprint
Fonte: Banco de imagens Shutterstock (2017)

6.2 Etapas do planejamento do Sprint

A **primeira atividade da reunião é definir a meta do Sprint**, ou seja, uma descrição sucinta do que deve ser entregue de valor para o cliente. Essa meta é definida pelo Product Owner e deve ser algo tangível, que possa ser avaliado pelo time. Além de tangível, a meta deve ser realista e aceita por todos os envolvidos no planejamento. Em hipótese alguma pode ser imposta.



Figura 6.4 – Analogia a reunião
Fonte: Banco de imagens Shutterstock (2017)

Com base na meta definida, o Product Owner escolhe os itens mais importantes do Product Backlog, que devem ser atualizados, detalhados e quebrados em histórias menores para que possam ser concluídas dentro do Sprint. Lembre-se de que:

- Embora, o Sprint tenha seu tamanho definido, nesse momento, ele pode sofrer mudanças para permitir que o time atinja a meta definida.
- O Product Backlog pode sofrer repriorizações ou inclusão de novas histórias.
- O tamanho das histórias pode sofrer alterações.
- O time Scrum define sua capacidade para Sprint, ou seja, qual é a sua velocidade.
- O Product Owner deve participar ativamente dessa fase com o time.

Na **sessão**, o Product Owner apresenta a meta do Sprint para o time, discute seu conteúdo e tira dúvidas, se for o caso. Depois, o time avalia e entende cada item definido do Product Backlog para verificar a possibilidade de entrega, dentro do prazo definido. Essa decisão deve ser baseada na capacidade do time (velocidade) e no conhecimento sobre as atividades a serem desenvolvidas.

IMPORTANTE: É essencial que o time realmente acredite que a meta possa ser cumprida e o comprometimento seja real e irrestrito com a execução das atividades.

Para cada item do Product Backlog, a equipe deve decompor em atividades e fazer a estimativa de quanto tempo (em horas) vai precisar para fazer tudo o que é necessário, a fim de concluir cada atividade do Sprint. O resultado final desse trabalho é chamado de **Sprint Backlog**. Na figura: “Fluxo do Planejamento do Sprint”, temos uma ilustração do fluxo dessas atividades.

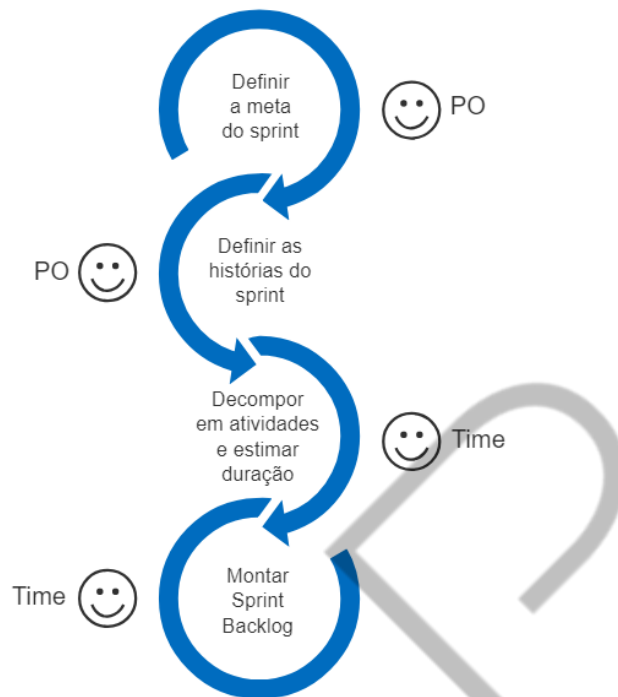


Figura 6.5 – Fluxo do Planejamento do Sprint
Fonte: Elaborado pelo autor, adaptado por FIAP (2017)

6.3 Sprint Backlog

O Sprint Backlog contém todas as atividades necessárias para completar as histórias selecionadas do Product Backlog. Todas as atividades devem ser estimadas, em horas, a fim de acompanhar a evolução e trabalho restante das atividades.

Normalmente, o Sprint Backlog utiliza o quadro *Kanban* no qual as atividades são colocadas em colunas, conforme mostra a figura: “Product Backlog no quadro Kanban”, contendo a descrição da atividade e o esforço em horas previsto pelo time.

O quadro Kanban é organizado em colunas que representam a ordem em que as tarefas são executadas, sendo as colunas mais comuns: **a fazer**, **fazendo** e **feito**. Porém, novas colunas podem ser inseridas pelo time, de acordo com as características do projeto ou do fluxo específico de cada organização.

IMPORTANTE: Um *Kanban* é um sistema criado, na Toyota, geralmente representado por um quadro, mas também organizado por meio de Software ou até mesmo uma folha de papel, em que cartões, que representam o trabalho, seguem um fluxo preestabelecido de estágios.

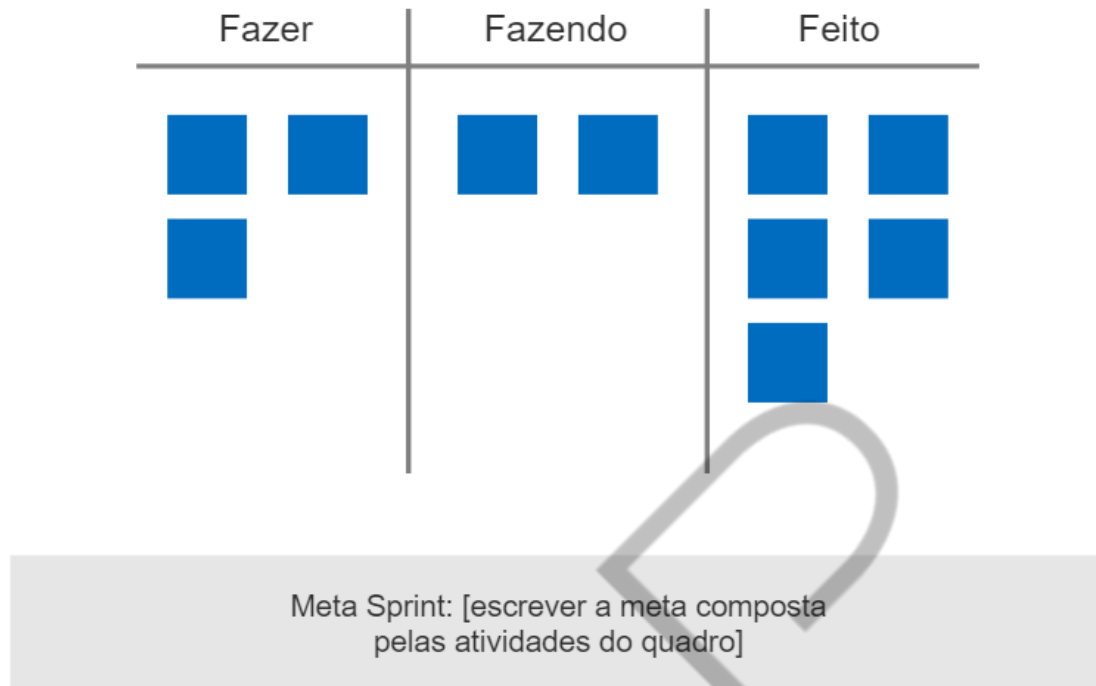


Figura 6.6 – Product Backlog no quadro Kanban
Fonte: Adaptado de cultura Ágil (2019)

No quadro Kanban não existe o conceito de dependências entre atividades, como em um cronograma. No entanto, a priorização é feita colocando as atividades de cima para baixo, ou colocando cores nos cartões, ou ainda, identificando, de alguma forma que fique claro para o time, o que deve ser feito.

Também é importante sempre identificar que uma atividade sofre mudança, ou uma nova é incluída, ou quando é preciso retornar um passo do fluxo de trabalho. O objetivo do Kanban é manter o time focado e melhorar a comunicação entre os envolvidos.

IMPORTANTE: Novas atividades podem ser adicionadas ao Sprint Backlog durante o Sprint, desde que não afetem o timebox definido, mas novas histórias não podem ser adicionadas.

6.3.1 Conceito de Ready

O objetivo do conceito de *Ready* (DoR) é verificar se a história está clara e se todos os membros do time têm uma compreensão do que precisa ser feito, ou seja, se a história tem informações suficientes para começar a ser construída. Cada organização, e/ou cada time, define quais são os requisitos necessários para considerar que a história está OK.



Figura 6.7 - Checklist
Fonte: Banco de imagens Shutterstock (2017)

Normalmente, o DoR é definido por meio de um *checklist* que será aplicado sobre cada história para fazer essa verificação. A seguir temos uma lista de verificações, que podem ser feitas em um DoR, com o time e o Product Owner (PO).

- A história tem um tamanho adequado para ser detalhada?
- A história pode ser construída dentro de um Sprint?
- As dependências entre as histórias estão identificadas?
- Existe um esboço do produto?
- Os critérios de aceitação estão descritos de forma clara e testável?
- Os requisitos não funcionais estão identificados?
- A história não tem impedimentos?

De posse desse checklist, o time pode verificar as condições de execução de uma história, para ajudar no momento do planejamento do Sprint, evitando que o timebox dessa reunião seja comprometido por uma série de problemas os quais podem ser previamente resolvidos.

IMPORTANTE: Da mesma forma que todos os artefatos do Scrum, o DoR pode ser revisado e atualizado para melhoria da qualidade e das estimativas de tempo. Essa revisão é feita na *Review Meeting*.

6.3.2 Reunião de Grooming

A palavra *Grooming* significa refinamento e, embora não esteja prevista no desenho do framework original do Scrum, é uma prática amplamente utilizada no mercado, dado que, no empirismo, foi verificado que, muitas vezes, as histórias estavam mal definidas e comprometiam o timebox, do planejamento do Sprint. Logo, o seu objetivo é melhorar a definição das histórias do Product Backlog para que esteja em um nível de maturidade melhor, antes da próxima reunião de planejamento do Sprint.

A Reunião de Grooming é realizada de dois a três dias, antes do final do Sprint corrente e, antes do início da próxima reunião de planejamento do Sprint, liderada pelo PO e com a participação de 10% a 50% do time, recomendado, respectivamente, pelos autores Mike Cohn (2000); Ken Schwaber e Sutherland (2016). A ideia central é revisar os itens do Product Backlog para esclarecer todos os pontos e dúvidas do time e evitar que o PO tenha que buscar informações de última hora.

A duração de uma Reunião de Grooming é um timebox de uma a duas horas para cada Sprint.

6.3.3 Conceito de PRONTO

O conceito de *pronto*, ou no inglês *definition of done* (DoD), é estabelecido pelo time Scrum para dar maior transparência ao processo de formalização que um Sprint está concluído. Também tem um formato de checklist e deve ser aplicado, antes de liberar o Sprint para revisão do PO.

O DoD deve ser definido e ter o comprometimento de todo o time Scrum e ter suas características publicadas a todos os envolvidos.



Figura 6.8 - Processos concluídos
Fonte: Banco de imagens Shutterstock (2017)

Um exemplo de checklist de DoD é definido a seguir:

- História definida, produto desenhado e validado pelo PO.
- História construída.
- Realizados os testes unitários sem erros.
- Funcionalidade inspecionada.
- Testes de aceitação 100% OK.

Sabe quando o cliente pergunta se está pronto? Como você sabe o que é pronto? DoD.

IMPORTANTE: A diferença entre o DoR e DoD é que o DoR é aplicado, antes de o Sprint iniciar, e o DoD é aplicado antes, de o Sprint terminar.

6.4 Testes de aceitação

A elaboração dos testes de aceitação também não é uma atividade do framework Scrum, porém é uma técnica que auxilia o time na validação das funcionalidades construídas, aumentando a qualidade do produto. Esses testes de aceitação devem ser escritos pelo Product Owner, com auxílio do Scrum Master, antes

da construção do produto, para permitir a avaliação preliminar do produto. A elaboração e execução dos testes de aceitação devem fazer parte do processo de elaboração do produto.

Vamos supor que temos a seguinte história:

Como um cliente cadastrado no site, posso pagar meu pedido com cartão de crédito.

Para essa história, elaboramos vários testes que garantem o correto funcionamento da história:

- Teste de pagamento com cartão vencido.
- Testes de pagamento com cartão sem limite para valor da compra.
- Teste de pagamento com cartão expirado.
- Testes de pagamento com as bandeiras Visa e Master.
- Testes de pagamento com mais de um cartão.
- Teste de pagamento com sucesso.
- Teste de pagamento com sucesso com mais de um cartão.

Essa especificação preliminar feita pelo PO permite que o time construa e verifique o produto, conforme os critérios de aceite estabelecidos, aumentando a qualidade do produto.

Podemos utilizar também o BDD (Behavior Driven Development) para auxiliar na elaboração dos testes, seguindo uma estrutura de linguagem natural que facilita o entendimento de todos.



Figura 6.9 - PO
Fonte: Banco de imagens Shutterstock (2017)

DICA: O BDD é uma técnica ágil para testes e consiste em ter as visões técnicas e de negócio juntas, permitindo identificar os comportamentos esperados do produto, em linguagem o mais natural possível. Veja mais em: <https://www.agilealliance.org/glossary/bdd/>

Como exemplo, para cada história do Product Backlog, escrevemos na seguinte estrutura:

Cenário: <descrição do teste>
Dado: <um estado conhecido>
Quando: <um determinado evento ocorre>
Então: <isso deve ocorrer>

Exemplo de uma descrição completa para uma história:

História:

Como um funcionário,

Eu gostaria de consultar um cliente cadastrado com sucesso por CPF,

De modo que eu possa garantir que a consulta está correta.

Caso de teste:

Cenário: consultar cliente com sucesso por CPF

Dado: ao selecionar a opção buscar

Quando: digitado o CPF 111.111.111-11

Então: deve listar nome, CPF e data de nascimento do cliente.

O uso do BDD facilita o entendimento dos usuários e a descrição mais completa dos critérios de testes, que devem ser executados.

6.5 Execução do Sprint

Durante a execução do Sprint, o time deve dar foco total ao que se comprometeu a entregar ao Product Owner. Nada pode atrapalhar o desempenho do time na execução do Sprint, momento no qual o Scrum Master atua fortemente como facilitador, fazendo a interface entre o time e o cliente, além de remover impedimentos e problemas para garantir a execução dos trabalhos. Esses impedimentos, muitas vezes, podem interromper uma atividade ou atrapalhar a evolução dela. Nesses casos, a atividade deve ser colocada, em situação de impedimento, e o Scrum deve atuar rapidamente.



Figura 6.10 – Execução do Sprint
Fonte: Banco de imagens Shutterstock (2017)

O Sprint Backlog é um artefato que deve ser atualizado, diariamente. Se um membro da equipe começa a trabalhar em uma atividade, seu nome está gravado dentro do Sprint Backlog. À medida que o trabalho vai evoluindo, os cartões vão mudando de coluna, e sempre que um novo trabalho é identificado, um novo cartão é criado, e o quadro Kanban é atualizado.

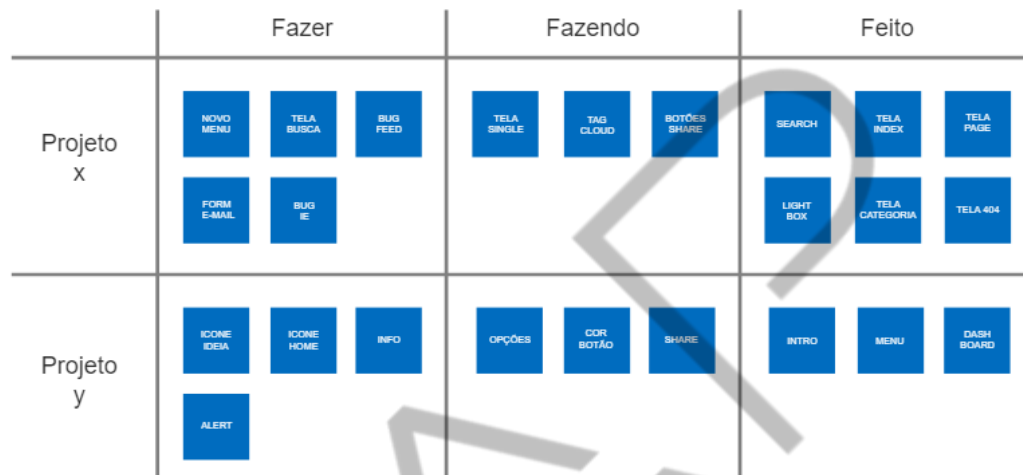


Figura 6.11 – Quadro Kanban durante a execução de um Sprint

Fonte: cultura Ágil, adaptado por FIAP (2017)

No final do dia, todas as atividades a concluir devem ter seus esforços atualizados, permitindo que o time avalie o quanto ainda falta do trabalho a ser feito e se está adiantado ou atrasado, gerando o Sprint Burndown, que veremos adiante.

Na execução do Sprint, as mudanças são bem-vindas, mas devem ser inseridas no Product Backlog para posterior planejamento. Isso é feito para evitar que o timebox do Sprint tenha seu prazo comprometido por novas demandas, fazendo com que o Product Owner não dê atenção ao plano, uma vez que ele pode alterar o Sprint quando quiser, além de desmotivar o time a cumprir os prazos. Isso seria o mesmo cenário dos projetos tradicionais...

6.5.1 Um Sprint pode ser cancelado?

Em princípio, um Sprint só pode ser cancelado em casos extremos, senão afeta os conceitos básicos de Scrum, de comprometimento e foco. Mas, se for extremamente necessário, o Sprint só pode ser cancelado em duas situações:

- O time verifica que as estimativas foram erradas e que não conseguirá entregar o combinado.
- O Product Owner percebe que mudanças relevantes ocorreram e afetaram a meta do Sprint.

Quando isso ocorre, os envolvidos devem iniciar imediatamente o planejamento do Sprint seguinte para evitar maiores impactos no projeto.

IMPORTANTE: Somente o Product Owner tem autoridade para cancelar um Sprint.

6.5.2 Reunião diária

A reunião diária é uma reunião de pé, de curta duração, realizada todos os dias, de preferência, durante o início do dia de trabalho, para melhorar a comunicação do time e permitir que o time tome ações corretivas, na execução dos trabalhos. Segundo o Scrum Guide®, durante essa reunião, cada membro da equipe deve fornecer brevemente as respostas para as três perguntas seguintes, sem desviar o foco:

- O que foi realizado desde a última reunião?
- O que vai realizar até a próxima reunião?
- Quais são os impedimentos em suas tarefas?

Todos os membros da equipe devem participar e eles devem ficar em pé, no local de trabalho, durante a reunião, em um timebox de **quinze minutos**.



Figura 6.12 – Reunião diária
Fonte: Banco de imagens Shutterstock (2017)

Os problemas ou dúvidas que a equipe não se considera apta a resolver são classificados como **Impedimentos** e passados para o **Scrum Master** para que ele resolva.

IMPORTANTE: A reunião diária faz parte do ciclo de inspeção e adaptação prevista nos pilares do Scrum.

6.5.3 Burndown Chart

O **Burndown Chart** é uma ferramenta de medição visual, que mostra o trabalho realizado por dia, contra a taxa projetada de conclusão para a versão atual do projeto. Sua finalidade é permitir a visualização do andamento do trabalho, com base na programação realizada. Pode acompanhar a evolução do projeto, do release ou do Sprint.

Sua principal característica é mostrar o quanto falta a ser feito, e não o que já foi feito, o que proporciona uma visão real do tamanho das atividades dentro do prazo restante. Tipicamente o Product Burndown é um gráfico que contém as informações do total de pontos a serem executados *versus* a quantidade de sprints, sendo a linha de velocidade opcional durante esse ciclo, conforme mostra a figura: “Product Burndown”.

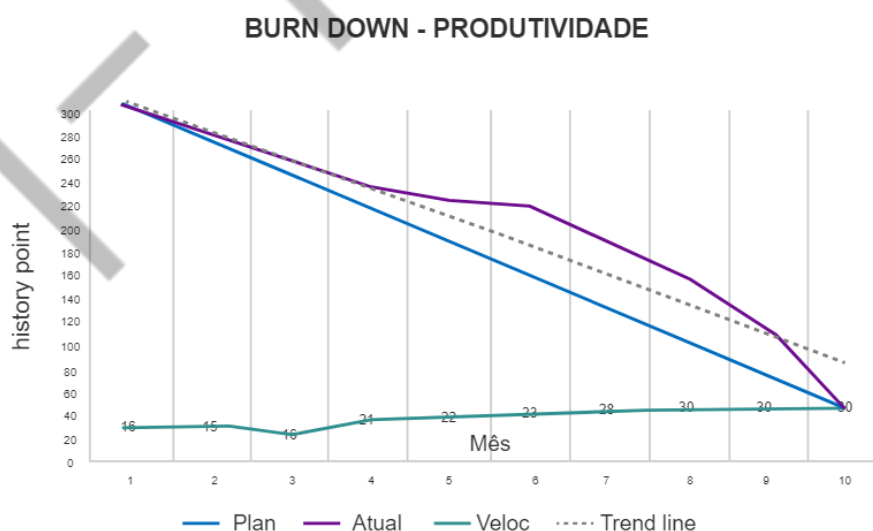


Figura 6.13 – Product Burndown
Fonte: Google Imagens (2017), adaptado por FIAP (2017)

Na figura: “Product Burndown”, podemos observar a evolução de um projeto com mil pontos de história que foi concluído em dez Sprints. A linha azul se refere ao

planejamento original de execução, a linha roxa mostra o comportamento do desempenho do time, ao longo do projeto, sendo observado que havia atrasos entre os sprints 3 e 9, mas que foi recuperado ao seu final. A linha pontilhada é a linha de tendência, que permite ao time tomar as ações de recuperação. A linha verde mostra a velocidade calculada pelo time, a cada sprint. Nota-se uma estabilização da velocidade, a partir do quarto Sprint, com uma leve melhoria gradual até o final do projeto.

O Sprint burndown, por ter características mais curtas, normalmente representa esforço *versus* duração, ou número de atividades *versus* duração, sendo o primeiro o mais comum. Na figura: “Sprint Burndown”, podemos observar um Sprint Burndown com dimensões em horas *versus* dias trabalhados, sendo um Sprint com um esforço previsto de 240 horas, em duas semanas (dez dias). Nesse Burndown, temos um desempenho do time, melhor do que o desempenho planejado, como podemos observar pela evolução da linha azul, em relação à linha de cor roxa, estando a Sprint no sexto dia de execução, com tendência de terminar, antes da data prevista.

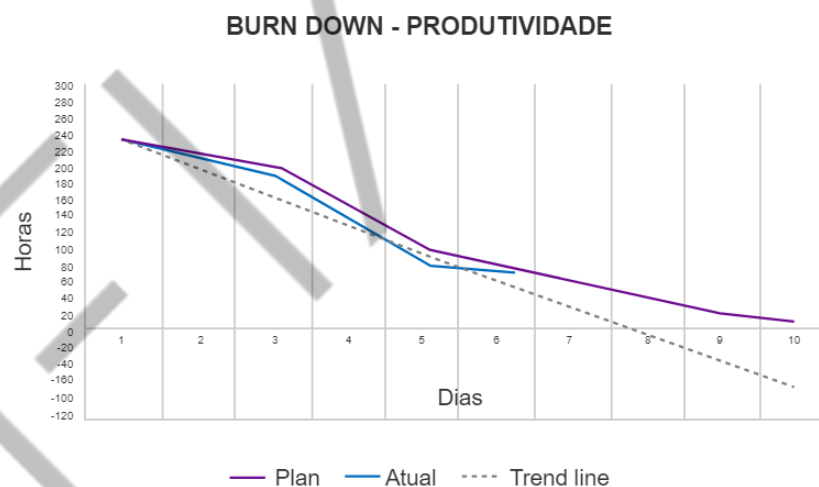


Figura 6.14 – Sprint Burndown
Fonte: Google Imagens (2017), adaptado por FIAP (2017)

Esse gráfico deve ser atualizado ao final do dia pelo time, de acordo com a evolução do quadro Kanban, para avaliação de desempenho e verificação de ações preventivas e corretivas.

6.6 Revisão do Sprint

Ao final do Sprint há a Revisão, da qual todos podem participar e tem como principal objetivo a avaliação do resultado do Sprint pelo Product Owner e os seus convidados.

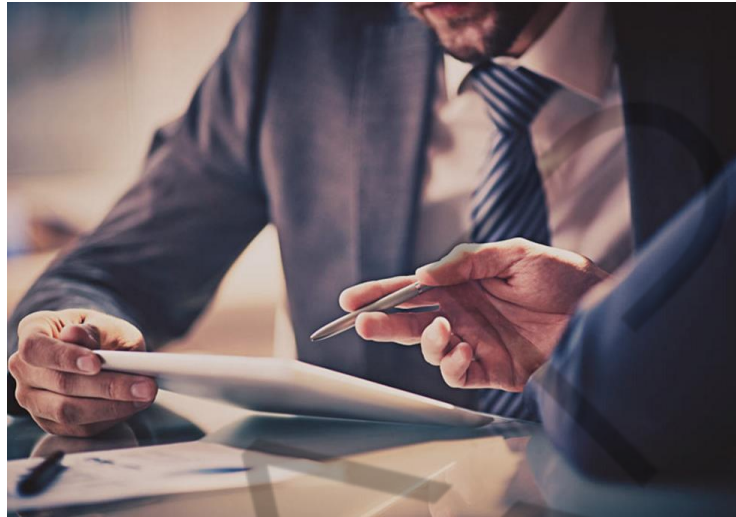


Figura 6.15 – Review Meeting
Fonte: Banco de imagens Shutterstock (2017)

O time é o responsável pela realização da apresentação, que precisa ser feita no formato de demonstração do resultado. Nessa reunião, o produto deve ser mostrado funcionando, conforme as especificações. Durante a *Review Meeting*, o Product Owner avalia se a meta do Sprint foi atingida, podendo:

- Aprovar integralmente o Sprint.
- Aprovar, com ressalvas, transformando as ressalvas em novos itens para o Product Backlog.
- Reprovar o Sprint e devolver todo o conteúdo ao Product Backlog.

A Review Meeting também é um timebox definido pelo Scrum:

- 4 horas para Sprint de quatro semanas.
- 3 horas para Sprint de três semanas.
- 2 horas para Sprint de duas semanas.

6.6.1 Resultados da revisão

Durante a reunião de revisão, além da atividade de avaliação, outras atividades são realizadas, segundo o Scrum Guide®:

- Retornar as funcionalidades não terminadas ao Product Backlog e repriorizá-las.
- Remover do Product Backlog funcionalidades que foram finalizadas.
- Trabalhar com o Scrum Master para reformular a equipe, caso necessário.
- Repriorização do Product Backlog.
- Solicitar o fechamento de um release com as funcionalidades demonstradas, sozinhas ou com o incremento de Sprints anteriores.
- Autorizar ou não autorizar o próximo Sprint.

6.6.2 Retrospectiva

Após a reunião de revisão, ocorre a reunião da retrospectiva do Sprint, com a participação do time Scrum e o Scrum Master, como facilitador. Nessa reunião todos os membros da equipe refletem sobre o Sprint passado e verificam três coisas:

- O que correu bem durante o Sprint?
- O que não correu bem durante o Sprint?
- Quais melhorias poderiam ser feitas para o próximo Sprint?



Figura 6.16 – Retrospectiva
Fonte: Banco de imagens Shutterstock (2017)

IMPORTANTE: Essa reunião é essencial para o processo de melhoria contínua do projeto.

É nela que o time se rearranja, estabelece novos padrões e corrige os problemas, permitindo a equipe melhorar a sua produtividade e a sua qualidade.

A retrospectiva também é um timebox, definido pelo Scrum:

- 3 horas para Sprint de quatro semanas.
- 2 horas para Sprint de três semanas.
- 1,5 hora para Sprint de duas semanas.

6.7 Dicas

Há uma grande resistência em utilizar Scrum em projetos grandes. Isso é possível, mas não quer dizer que seja fácil de ser implementado.

O conceito para trabalhar com grandes times é chamado de Scrum of Scrums, pois no Scrum temos limite de times com até nove pessoas. Portanto, quando falamos em escalar o Scrum, estamos falando de criar tantos times de nove pessoas, quantos forem necessários para atender à demanda do projeto.

Nesse cenário de vários times trabalhando, ao mesmo tempo, cada time terá seu Scrum Master e seu Product Owner, mas podemos ter vários problemas decorrentes disso, entre eles:

- Conflitos de liderança;
- falhas de comunicação;
- planejamento descentralizado;
- duplicação de esforço e retrabalho;
- problemas com a integração dos produtos;
- dependências entre as atividades;
- difícil controle de mudanças.

Para solucionar esses problemas, existem várias sugestões, inclusive, um guia específico só para tratar desse assunto, chamado: *Nexus Guide*. Vamos listar alguns tópicos que ajudam na solução de alguns itens:

- Dividir o escopo do projeto em incrementos mais independentes possíveis entre si, limitando as dependências entre os times.
- Criar papéis para centralização de decisões no nível do Scrum Master e do Product Owner.
- Criar um Product Backlog único.
- Criar marcos (pontos de controle) para unificação de releases e publicação de uma nova versão.
- Fazer reuniões diárias conjuntas para que representantes de outros times possam participar, com o objetivo de compartilhar problemas, dependências e conhecimento.
- Criar uma reunião de Scrum de Scrums para o acompanhamento das atividades, verificação de dependências e centralização das mudanças.

Não é objetivo aqui esgotarmos esse assunto, mas sim passar uma visão geral e pontos críticos, que devem ser observados em eventual necessidade de utilizar vários times.

IMPORTANTE: Para mais detalhes, para escalar o SCRUM, consulte o *Nexus Guide*, no site: <www.scrum.org>.

A seguir algumas ferramentas gratuitas:

- **ScrumHalf** – Brasileira, e simplifica o uso do Scrum. Tem um quadro Kanban virtual e facilita a colaboração e o acompanhamento da equipe. Fácil dar manutenção e priorizar o Product Backlog; gera o Burn Down Chart, entre outros: <<http://myscrumhalf.com/?lang=pt>>.
- **Pango Scrum** – Brasileira, e com ela é possível fazer todo gerenciamento do Product Backlog, planejar os Sprints e monitorar o progresso. Não tem quadro Kanban, deve-se usar o modelo manual: <<http://pangoscrum.com/pt-BR>>.
- **Trello** não é uma ferramenta de Scrum, mas automatiza o quadro Kanban: <<https://trello.com/>>.

REFERÊNCIAS

AGILE ALLIANCE. **Behavior Driven Development (BDD)**. Disponível em: <<https://www.agilealliance.org/glossary/bdd/>>. Acesso em: 14 jun. 2020.

COHN, M. **Desenvolvimento com Scrum: Aplicando métodos ágeis com sucesso**. Editora: Cidade, 2000. GLOSSARY.

EQUIPE CULTURA ÁGIL. **Management 3.0 – O método ágil de gestão empresarial**. Disponível em: <<https://www.culturaagil.com.br/management-3-0-o-metodo-agil-de-gestao-empresarial/>>. Acesso em: 14 jun. 2020.

SCHWABER, K.; SUTHERLAND, J. **Scrum Guide**. Scrum Org, 2016.

SCHWABER, K.; SUTHERLAND, J. **Nexus Guide**. Scrum Org, 2016.

SCHWABER, K.; SUTHERLAND, J. **Software em 30 dias**. John Wiley, 2012.

SCRUM. **Scrum Website**. [s.d.]. Disponível em: <<http://www.scrum.org>>. Acesso em: 14 jun. 2020.

SCRUM ALLIANCE. **ScrumAlliance Website**. [s.d.]. Disponível em: <<https://www.scrumalliance.org/>>. Acesso em: 14 jun. 2020.

WEST, D. **Updates to the Scrum Guide - The 5 Scrum values take center stage**. Publicado em: 06 jul. 2016. Disponível em: <<https://blog.scrum.org/updates-scrum-guide-5-scrum-values-take-center-stage/>>. Acesso em: 14 jun. 2020.