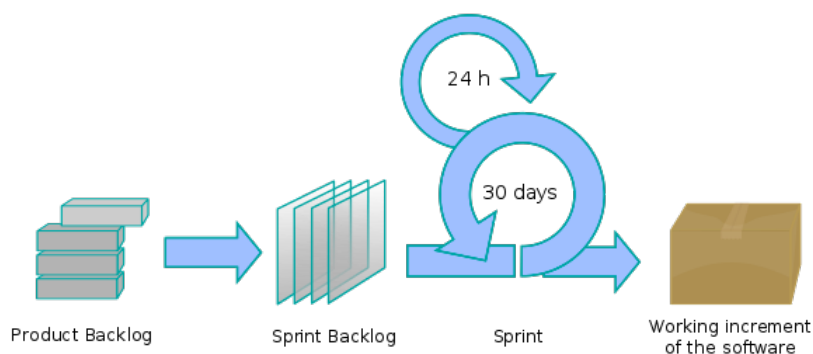


Scrum (desenvolvimento de software)

Origem: Wikipédia, a enciclopédia livre.

O **Scrum** (pron. [skram]) é um framework de gerenciamento de projetos, da organização ao desenvolvimento ágil de produtos complexos e adaptativos com o mais alto valor possível, através de várias técnicas, utilizado desde o início de 1990 e que atualmente é utilizado em mais de 60% dos projetos ágeis em todo o mundo. ^{[1][2]}



Representação visual dos principais artefatos do Scrum Framework e seu relacionamento com a Sprint

Este framework não é um processo linear, em vez disso, é um conjunto de conceitos e técnicas dentro do qual você pode empregar vários outros processos ou técnicas. Analisando a eficácia relativa das práticas de gerenciamento e desenvolvimento, de modo que você possa melhorá-las. Este também não é um processo prescribente, ou seja, não descreve o que fazer em cada situação, pois em trabalhos complexos é impossível prever tudo o que irá ocorrer. É um conjunto de valores, princípios e práticas que fornecem a base para que a sua organização adicione suas práticas particulares de engenharia e gestão e que sejam relevantes para a realidade da sua empresa. O resultado será uma versão de Scrum que é exclusivamente sua. ^[3]

Scrum possui seu foco no gerenciamento e projeto da organização onde é difícil planejar à frente. Mecanismos do Controle de Processo Empírico, onde ciclos de feedback constituem o núcleo da técnica de gerenciamento que são usadas em oposição ao tradicional gerenciamento de comando e controle. ^[4] É uma forma de planejar e gerenciar projetos trazendo a autoridade da tomada de decisão a níveis de propriedade de operação e certeza. ^[5]

Apesar da palavra não ser um acrônimo, algumas empresas que implementam o processo a soletram com letras maiúsculas como SCRUM. Isto pode ser devido aos primeiros artigos de Ken Schwaber, que capitalizava SCRUM no título.

Apesar de ser criado para gerenciamento de projetos de software, também pode ser utilizado em equipes de manutenção de software ou como uma abordagem geral de gerenciamento de projetos/programas.

Atualmente, as técnicas de DevOps são utilizados por programadores no desenvolvimento de software em conjunto com técnicas de gestão e desenvolvimento ágil de software, como o Scrum.^{[6][7][8]}

Índice

História

O Guia do Scrum

Manutenção e Evolução do Guia do Scrum

Características do Scrum

Gerenciamento Ágil de Projetos com Scrum

Sprint

Papéis

Papéis principais

Papéis auxiliares

Artefatos Oficiais do Scrum^[4]

Product Backlog

Sprint Backlog

Artefatos Complementares ao Scrum

Burndown Chart

Eventos Scrum

Scrum Solo

Desempenho da equipe e Escopo de Utilização

Referências

Ver também

Ligações externas

História

Inicialmente, o Scrum foi concebido como um estilo de gerenciamento de produtos em empresas de fabricação de automóveis e produtos de consumo, por Takeuchi e Nonaka no artigo "The New Product Development Game" (Harvard Business Review, Janeiro-Fevereiro 1986). Eles notaram que projetos usando equipes pequenas e multidisciplinares (*cross-functional*) produziram os melhores resultados, e associaram estas equipes altamente eficazes à formação Scrum do Rugby (utilizada para reinício do jogo em certos casos). Jeff Sutherland,^[9] John Scumniotales e Jeff McKenna conceberam, documentaram e implementaram o Scrum, conforme descrito abaixo, na empresa Easel Corporation em 1993, incorporando os estilos de gerenciamento observados por Takeuchi e Nonaka.

Ken Schwaber e Jeff Sutherland fizeram a primeira co-apresentação do Scrum na

conferência OOPSLA de 1995. Esta apresentação essencialmente documentou o aprendizado que Ken e Jeff tiveram ao longo dos anos anteriores na aplicação do Scrum. e ajudaram a implantá-lo no desenvolvimento de softwares em todo o mundo.

Scrum junta conceitos de Lean, desenvolvimento iterativo e do estudo de Hiroataka Takeuchi e Ikujiro Nonaka.

A função primária do Scrum é ser utilizado para o gerenciamento de projetos de desenvolvimento de software. Ele tem sido usado com sucesso para isso, assim como Extreme Programming e outras metodologias de desenvolvimento. Porém, teoricamente pode ser aplicado em qualquer contexto no qual um grupo de pessoas necessitem trabalhar juntas para atingir um objetivo comum, como iniciar uma escola pequena, projetos de pesquisa científica, ou até mesmo o planejamento de um casamento.^[1]

Mesmo que idealizado para ser utilizado em gestão de projetos de desenvolvimento de software ele também pode ser usado para a gerência de equipes de manutenção, ou como uma abordagem para gestão de programas: *Scrum de Scrums*.

O Guia do Scrum

O Guia do Scrum é o documento oficial que documenta o Scrum, suas regras e definições conforme desenvolvido e sustentado por mais de 20 anos por Jeff Sutherland e Ken Schwaber.

O documento é mantido por duas instituições, a Scrum.Org e seus Professional Scrum Trainers (PSTs) e a Scrum Alliance. Ambas oferecem o cursos preparatórios para certificações relacionadas ao Scrum.^[10]

Manutenção e Evolução do Guia do Scrum

Os criadores do Scrum Ken Schwaber e Jeff Sutherland são os responsáveis primários por toda e qualquer alteração direta ao documento do Guia do Scrum. Além disso os Professional Scrum Trainers, ou PST's, também atuam na evolução do Framework Scrum e no fornecimento de sugestões de melhorias para o documento e sua aplicação no mundo real. Sua atuação em empresas e com a comunidade é imprescindível para a evolução dos métodos, processos e ferramentas ágeis e no uso e adoção do Scrum.

O "*Professional Scrum Trainer*" (PST) é um título concedido pela Scrum.org a profissionais que demonstram profundos conhecimentos práticos e teóricos sobre o Framwork Scrum e que são aprovados em um rigoroso e extenso processo seletivo para tal. O título de PST representa a mais alta honraria e reconhecimento dentro da comunidade Scrum e seus portadores são reconhecidos como verdadeiras autoridades na adoção, implementação e uso do Scrum e suas práticas complementares em empresas, governos e organizações.^[11]

Atualmente existem 6 diferentes tipos de PST, cada um com uma diferente especialidade dentro do universo Scrum. Um PST pode ter uma ou mais especialidades e ser reconhecido por elas, que são:

- **Professional Scrum Master:** PST especializado nos processos e ferramentas relacionados ao papel de Scrum Master;
- **Professional Scrum Product Owner:** PST especializado nos processos e ferramentas relacionados ao papel de Scrum Master;
- **Professional Scrum Developer:** PST especializado nos processos e ferramentas relacionados ao papel de Scrum Master;
- **Scaled Professional Scrum:** PST especializado nos processos e ferramentas relacionados ao papel de Scrum Master;
- **Professional Scrum with Kanban:** PST especializado nos processos e ferramentas relacionados ao papel de Scrum Master;
- **Professional Agile Leadership:** PST especializado nos processos e ferramentas relacionados ao papel de Scrum Master;

Devido à longa duração e do alto rigor do processo para reconhecimento dos PST's existe um número reduzido de profissionais que possuem este título em todo o mundo. Atualmente existem 291 pessoas ao redor do globo que são PST's de um ou mais temas. Destes, apenas 7 pertencem à países lusófonos e estão todos concentrados no Brasil. Os 7 PST's no Brasil são:

- Alexandre Mac Fadden
- André Coelho
- André Gomes (<http://andrelmgomes.com.br>)
- Antonio Costa Neto
- Ivan Jorge Vieira dos Santos
- Rodrigo Silva Pinto
- Victor Hugo de Oliveira

Características do Scrum

- Clientes tornam-se parte da equipe de desenvolvimento (os clientes devem estar genuinamente interessados na saída);
- Entregas frequentes e intermediárias de funcionalidades 100% desenvolvidas;
- Planos frequentes de mitigação de riscos desenvolvidos pela equipe;
- Discussões diárias de status com a equipe de desenvolvimento;
- A discussão diária na qual cada membro da equipe de desenvolvimento responde às seguintes perguntas:
 - O que fiz desde ontem em direção a meta?
 - O que estou a pensar fazer até amanhã em direção a meta?
 - Existe algo que me impede de atingir a meta?
- Transparência no planejamento e desenvolvimento;
- Reuniões frequentes com os *stakeholders* (partes interessadas no projeto) para

monitorar o progresso;

- Problemas não são ignorados e ninguém é penalizado por reconhecer ou descrever qualquer problema não visto;
- Locais e horas de trabalho devem ser energizadas, no sentido de que "trabalhar horas extras" não necessariamente significa "produzir mais".

Gerenciamento Ágil de Projetos com Scrum

Scrum não só reforçou o interesse em gerenciamento de projetos de software, mas também desafiou as ideias convencionais sobre essa gestão. Scrum é voltado para instituições de gerenciamento de projetos, onde é difícil planejar o futuro com mecanismos de controle de processos empíricos, como loops de feedback, onde constituem o elemento central do desenvolvimento do produto em comparação com a gestão de comando e controle tradicionais orientado. Ela representa uma abordagem radicalmente nova para o planejamento e gerenciamento de projetos de software, trazendo poder de decisão ao nível das propriedades operação e certezas. Scrum reduz defeitos e torna o processo de desenvolvimento mais eficiente, bem como reduzindo os custos de manutenção a longo prazo.

Sprint

Uma sprint é a unidade básica de desenvolvimento em Scrum. Sprints tendem a durar entre uma semana e um mês, e são um esforço dentro de uma faixa de tempo (ou seja, restrito a uma duração específica) de comprimento constante. A adoção de ciclos relativamente curtos permite entregas de partes dos sistemas, gerando valor para os clientes e permitindo uma avaliação dinâmica do trabalho.

Cada sprint é precedida por uma reunião de planejamento (Sprint Planning), onde as tarefas para a sprint são identificadas e um compromisso estimado para o objetivo da sprint é definido, e seguido por uma reunião de revisão ou de retrospectiva, onde o progresso é revisto e lições para as próximas sprints são identificadas.

Durante cada sprint, a equipe cria um incremento de produto potencialmente entregável (por exemplo, software funcional e testado). O conjunto de funcionalidades que entram em uma sprint vêm do Product Backlog, que é um conjunto de prioridades de requisitos de alto nível definidos pelo **Product Owner**.

Quais itens do backlog que entram para a sprint são determinados durante a reunião de planejamento da sprint (Sprint Planning). Durante esta reunião, o **Product Owner** informa a equipe dos itens no backlog do produto que ele ou ela quer concluídos.

A equipe então determina quantos eles podem se comprometer a concluir durante a próxima sprint, e registram isso no backlog da sprint. Durante uma sprint, ninguém está autorizado a alterar o backlog da sprint, o que significa que os requisitos são congelados para essa sprint.

O desenvolvimento de cada sprint deve terminar na "caixa de tempo" prevista. Se os requisitos não são completados por qualquer motivo, eles são deixados de fora e voltam para o backlog do produto. Depois que uma sprint é completada, a equipe demonstra como usar o software.

O Scrum permite a criação de equipes auto-organizadas, encorajando a co-localização de todos os membros da equipe e a comunicação verbal entre todos os membros e disciplinas da equipe no projeto.

Um princípio chave do Scrum é o reconhecimento de que, durante um projeto, os clientes podem mudar de ideia sobre o que eles querem e precisam (muitas vezes chamados requisitos churn), e que os desafios imprevisíveis não podem ser facilmente tratados de uma maneira preditiva ou planejada tradicional. Como tal, o Scrum adota uma abordagem empírica, aceitando que o problema não pode ser totalmente entendido ou definido, focando na maximização da habilidade da equipe para entregar rapidamente e responder às necessidades emergentes.

Como outras metodologias de desenvolvimento ágil, o Scrum pode ser implementado através de uma ampla gama de ferramentas. Muitas empresas utilizam ferramentas de software universais, como planilhas para construir e manter artefatos como o backlog da sprint. Há também pacotes de software open-source e proprietários dedicados à gestão de produtos no âmbito do processo Scrum. Outras organizações implementam o Scrum sem o uso de quaisquer ferramentas de software, e mantêm seus artefatos na forma de cópias impressas, como papel, quadros e notas.

- Cada *sprint* é uma iteração que segue um ciclo (PDCA) e entrega incremento de software pronto.
- Um *backlog* é conjunto de requisitos, priorizado pelo Product Owner (responsável pelo ROI e por conhecer as necessidades do cliente);
- Há entrega de conjunto fixo de itens do *backlog* em série de interações curtas ou *sprints*;
- Breve reunião diária, ou daily scrum, em que cada participante fala sobre o progresso conseguido, o trabalho a ser realizado e/ou o que o impede de seguir avançando (também chamado de Standup Meeting ou Daily Meeting, já que os membros da equipe geralmente ficam em pé para não prolongar a reunião).
- Breve sessão de planejamento, na qual os itens do *backlog* para uma *sprint* (iteração) são definidos;
- Revisão da Sprint, é executada no final da Sprint, para inspecionar o incremento e adaptar o backlog do produto se necessário. As partes interessadas colaboram sobre o que foi feito na Sprint e com base nisso, e em qualquer mudança no backlog do produto durante a sprint, os participantes colaboram naquilo que pode ser feito para otimizar o valor do produto.
- Retrospectiva, é uma oportunidade para o Time Scrum inspecionar a si próprio e criar um plano para melhorias a serem aplicadas na próxima Sprint.

O Scrum é facilitado por um *Scrum Master*, que tem como função primária remover qualquer impedimento à habilidade de uma equipe de entregar o objetivo da *sprint*. O Scrum Master não é o líder da equipe (já que as equipes são

auto-organizadas), mas atua como um mediador entre a equipe e qualquer influência desestabilizadora. Outra função extremamente importante de um Scrum Master é o de assegurar que a equipe esteja utilizando corretamente as práticas de Scrum, motivando-os e mantendo o foco na meta da Sprint.

Papéis

Scrum é um esqueleto de processos que contém grupos de práticas e papéis pré-definidos. Os três papéis definidos no Guia Scrum [\[carece de fontes?\]](#) são:

1. o **Scrum Master**, que mantém os processos (normalmente no lugar de um gerente de projeto);
2. o **Dono do Produto**, ou **Product Owner**, que representa os *stakeholders* e o negócio;
3. a **Equipe de desenvolvimento**, ou **DevTeam**, um grupo multifuncional entre 3 a 9 pessoas e que fazem a análise, projeto, implementação, teste etc.

Papéis principais

Os papéis principais em equipes Scrum são aqueles comprometidos com o projeto no processo do Scrum - são os que produzem o produto (objetivo do projeto).

Product Owner (dono do produto)

O Product Owner representa a voz do cliente e é responsável por garantir que a equipe agregue valor ao negócio. O Product Owner escreve centrado nos itens do cliente (histórias tipicamente do usuário), os prioriza e os adiciona para o product backlog. Equipes de Scrum devem ter um Product Owner, e, embora esse possa também ser um membro da equipe de desenvolvimento, recomenda-se que este papel não seja combinado com o de ScrumMaster..

Equipe de Desenvolvimento (*Development Team*)

A equipe de desenvolvimento é responsável pela entrega do produto. A equipe é tipicamente composta de 5-9 pessoas com habilidades multifuncionais que fazem o trabalho real (analisar, projetar, desenvolver, testar técnicas de comunicação, documentos, etc.) Recomenda-se que a equipe seja auto-organizada e auto-conduzida, mas que muitas vezes trabalhem com alguma forma de projeto ou gestão de equipe.

Scrum Master

Scrum é facilitado por um Scrum Master, que é responsável pela remoção de impedimentos à capacidade da equipe para entregar o objetivo da sprint / entregas. O Scrum Master não é o líder da equipe, mas age como um tampão entre a equipe e qualquer influência ou distração. O Scrum Master garante que o processo Scrum seja usado como pretendido. O Scrum Master é o responsável pela aplicação das regras. Uma parte fundamental do papel do Scrum Master é proteger a equipe e mantê-la focada nas tarefas em mãos. O papel também tem sido referido como um líder-servo para reforçar essa

dupla perspectiva.

Papéis auxiliares

Os papéis auxiliares em equipes Scrum são aqueles com nenhum papel formal e envolvimento frequente no processo de Scrum, mas, ainda assim, devem ser levados em conta.

Partes interessadas (clientes, fornecedores)

Estas são as pessoas que permitem o projeto e para quem o projeto vai produzir o acordado benefício, que justifica a sua produção. Eles só estão diretamente envolvidos no processo durante as revisões sprint.

Gerentes (incluindo gerentes de projeto)

Pessoas que irão configurar o ambiente para desenvolvimento de produtos.

Artefatos Oficiais do Scrum^[4]

O Scrum Guide define os seguintes artefatos obrigatórios como necessários para o sucesso de um projeto que utilize Scrum:

Product Backlog

O *Product Backlog* é mantido pelo *Product Owner* e é uma lista de requisitos que tipicamente vêm do cliente. O *Product Backlog* pode ser alterado a qualquer momento pelo *Product Owner* ou por decisão deste.

Um *backlog* é uma lista de itens priorizados a serem desenvolvidos para um software. Este artefato é a principal fonte de informação para o Planejamento de *sprint* (Sprint Planning). No decorrer da *sprint*, o *Product Owner*, o *Scrum Master* e a Equipe decidem no que a equipe irá trabalhar. O *Product Owner* mantém uma lista priorizada de itens de *backlog*, o *backlog* do produto, o que pode ser repriorizado durante o planejamento da *sprint*. A Equipe seleciona itens do topo do *backlog* do produto. Eles selecionam somente o quanto de trabalho eles podem executar para terminar. A Equipe então planeia a arquitetura e o design de como o *backlog* do produto pode ser implementado. Os itens do *backlog* do produto são então destrinchados em tarefas que se tornam o *backlog* da *sprint*.

Sprint Backlog

O *Sprint Backlog* é uma lista de itens selecionados do *Product backlog* e contém tarefas concretas que serão realizadas durante a próxima *sprint* para implementar tais itens selecionados.

O *Sprint Backlog* é uma representação em tempo real do trabalho que o *Development Team* planeia concluir na *sprint* corrente, e ele pertence unicamente ao *Development Team*.

Artefatos Complementares ao Scrum

Além dos artefatos oficiais estabelecidos no Scrum Guide,^[4] existem artefatos complementares que podem ser utilizados para melhorar a qualidade das entregas e a eficiência do time. Alguns deles são:

Burndown Chart

O Burndown é um simples gráfico, com dois eixos X e Y, baseado nas atividades que não ultrapassem um dia de trabalho. O eixo Y indica o número de tarefas existentes na sprint e o eixo X os dias que representam o tamanho da sprint.

Eventos Scrum

Um momento bom para as discussões diárias é depois do almoço. Durante a manhã pode ser complicado. *[carece de fontes?]* Estas discussões de status não demoram e uma forma eficiente de fazer estas reuniões seria ficar em pé e em frente a um quadro negro. Como as pessoas tendem a ficar cansadas depois do almoço, ter uma viva reunião em pé nessa hora permite que a equipe mantenha a sua energia alta. Como todos estiveram trabalhando durante a manhã, suas mentes estão focadas no trabalho e não em questões pessoais. Grandes usuários do processo são enfáticos na necessidade de os membros da equipe estarem em pé durante a reunião, para permitir maior agilidade e evitar perdas no foco. Recomenda-se inclusive evitar lugares onde as pessoas possam se apoiar. *[carece de fontes?]*

Daily Scrum Meeting

Cada dia durante a sprint, uma reunião de status do projeto ocorre. Isso é chamado de "scrum diário", "daily scrum" ou "Stand up meeting" (reunião em pé). Esta reunião tem diretrizes específicas:

- A reunião começa precisamente no horário marcado.
- Todos são bem-vindos, mas apenas "poucos" podem falar.
- O encontro tem duração determinada (Time-Box) e dura no máximo 15 minutos.
- A reunião deve acontecer no mesmo local e mesma hora todos os dias
- Durante a reunião, cada membro da equipe responde a três perguntas:
 - O que você tem feito desde ontem em direção a meta?
 - O que você está planejando fazer hoje em direção a meta?
 - Você tem algum problema impedindo você de realizar seu objetivo em direção a meta?

É papel do Scrum Master facilitar a resolução desses impedimentos. Normalmente, isso deve ocorrer fora do contexto do Daily Scrum para que a reunião possa durar menos de 15 minutos.

Reunião de planejamento da Sprint (*Sprint Planning Meeting*)

No início do ciclo de sprint, um *Sprint Planning Meeting* é realizado.

- Selecione o trabalho que está a ser feito.
- Prepare o Sprint Backlog que detalha o tempo que levará para fazer esse trabalho, com toda a equipe.
- Identificar e comunicar o quanto o trabalho é susceptível de ser feito durante a sprint atual.
- Dividida em duas partes:
 - Parte 1 (Primeiras quatro horas): Team Product Owner: diálogo para priorizar o Product Backlog.
 - Parte 2 (Próximas quatro horas): Team apenas: hash de um plano para a Sprint, resultando na Sprint Backlog.

No final de um ciclo de sprint, são realizadas duas reuniões: a "Sprint Review" e do "Sprint Retrospective".

Reunião de Revisão da Sprint (*Sprint Review*)

- Rever o trabalho que foi concluído e não concluído.
- Apresentar o trabalho realizado para os *stakeholders* (ou "a demo"). Um trabalho incompleto não pode ser demonstrado.

Retrospectiva da Sprint (*Sprint Retrospective*)

- Todos os membros da equipe refletem sobre a sprint passada.
- Faça melhorias contínuas de processos.

Duas questões principais são feitas na retrospectiva da sprint: O que correu bem durante a corrida? O que poderia ser melhorado na próxima sprint?

Scrum Solo

Scrum é baseado em pequenas equipes. Ele permite a comunicação entre os membros da equipe. Entretanto, há uma grande quantidade de softwares desenvolvidos por programadores solos. Um software sendo desenvolvido por um só programador pode ainda se beneficiar de alguns princípios do Scrum, como: um *backlog* de produto, um *backlog* de *sprint*, um *sprint* e uma retrospectiva de *sprint*. Scrum Solo é uma versão adaptada para uso de programadores solo.^[12]

Desempenho da equipe e Escopo de Utilização

O desempenho da equipe será definido com base no histórico de entregas e conclusões de projetos anteriores, ações que iram definir o quanto em pontos a equipe consegue atingir. Através de um sistema de pontuação cada desenvolvedor atinge em média 20 pontos por sprint de duração de duas semanas. Tal desempenho pode ser afetado por diversos fatores, os quais que devem ser trazidos para discussão na reunião de retrospectiva e planejamento.

Nesta reunião, deve-se abordar a apresentação de problemas e dificuldade encontrados, buscando assim resoluções, melhorias no processo 16 de desenvolvimento de software e conseqüentemente objetivando a velocidade de conclusão dos próximos sprints [13].

O Scrum é uma metodologia destinada a pequenas equipes com menos de dez pessoas. Schwaber e Beedle [14] apud [15] sugerem que a equipe seja composta de cinco a nove integrantes, se mais pessoas estiverem envolvidas no projeto, devem-se formar múltiplas Equipes Scrum.[16]

Referências

1. «Scrum: o Framework Mínimo Viável» (<https://www.linkedin.com/pulse/scrum-o-framework-m%C3%ADnimo-vi%C3%A1vel-andr%C3%A9-gomes>). *www.linkedin.com*. Consultado em 30 de setembro de 2019
2. «14th Annual State of Agile Survey» (<https://www.stateofagile.com/#ufh-c-473508-state-of-agile-report>). *CollabNet VersionOne*. Consultado em 1 de outubro de 2019
3. «Quero usar o Scrum. Por onde começo?» (<https://www.linkedin.com/pulse/quer-o-aplicar-o-scrum-por-onde-come%C3%A7o-andr%C3%A9-gomes>). *www.linkedin.com*. Consultado em 25 de setembro de 2019
4. «Home | Scrum Guides» (<http://www.scrumguides.org>). *www.scrumguides.org* (em inglês). Consultado em 1 de junho de 2017
5. Schwaber, Ken (fevereiro de 1, 2004). *Agile Project Management with Scrum*. [S.l.]: Microsoft Press. ISBN 978-0-7356-1993-7
6. DevOps and Agile (Scrum Alliance) (<https://www.scrumalliance.org/community/articles/2014/april/devops-and-agile>)
7. Qual a diferença entre extreme programming, Scrum e DevOps? (<https://gaea.com.br/qual-a-diferenca-entre-extreme-programming-scrum-e-devops/>)
8. Entenda o que a união entre DevOps e Scrum pode gerar (<https://gaea.com.br/entenda-o-que-a-uniao-entre-devops-e-scrum-pode-gerar/>)
9. *Scrum - a arte de fazer o dobro de trabalho na metade do tempo*. [S.l.: s.n.] 2014 | nome1= sem | sobrenome1= em Authors list (ajuda)
10. Site do Scrum.Org oferece o curso de treinamento oficial de Scrum Master (<http://www.scrum.org/>)
11. «Become a Professional Scrum Trainer» (<https://www.scrum.org/become-professional-scrum-trainer>). *Scrum.org* (em inglês). Consultado em 1 de outubro de 2019
12. Pagotto, T.; J. A. (1 de junho de 2016). «Scrum solo: Software process for individual development» (<http://ieeexplore.ieee.org/document/7521555/>). 2016 *11th Iberian Conference on Information Systems and Technologies (CISTI)*: 1–6. doi:10.1109/CISTI.2016.7521555 (<https://dx.doi.org/10.1109%2FCISTI.2016.7521555>)

3. Faoro, Mariana Wentz; Olinto, Maria Teresa Anselmo; Paniz, Vera Maria Vieira; Macagnan, Jamile; Henn, Ruth Liane; Garcez, Anderson; Pattussi, Marcos Pascoal (2018). «Dor musculoesquelética relacionada ao trabalho e sua associação com transtornos mentais comuns em trabalhadores de um frigorífico do Sul do Brasil» (<https://dx.doi.org/10.5327/z1679443520180200>). *Revista Brasileira de Medicina do Trabalho*. **16** (2): 136-144. ISSN 1679-4435 (<https://www.worldcat.org/issn/1679-4435>). doi:10.5327/z1679443520180200 (<https://dx.doi.org/10.5327%2Fz1679443520180200>)
4. Julnes, G (2001). «Evaluation (2nd Edition), by Carol Hirshon Weiss, Upper Saddle River, NJ: Prentice Hall, 1998, 372 pp.» ([https://dx.doi.org/10.1016/s1098-2140\(01\)00123-0](https://dx.doi.org/10.1016/s1098-2140(01)00123-0)). *The American Journal of Evaluation*. **22** (2): 265-268. ISSN 1098-2140 (<https://www.worldcat.org/issn/1098-2140>). doi:10.1016/s1098-2140(01)00123-0 (<https://dx.doi.org/10.1016%2Fs1098-2140%2801%2900123-0>)
5. Franco, Eduardo Ferreira. «Um modelo de gerenciamento de projetos baseado nas metodologias ágeis de desenvolvimento de software e nos princípios da produção enxuta.» (<https://dx.doi.org/10.11606/d.3.2007.tde-09012008-155823>)
6. «Scrum (desenvolvimento de software)» ([https://pt.wikipedia.org/w/index.php?title=Scrum_\(desenvolvimento_de_software\)&oldid=55061933](https://pt.wikipedia.org/w/index.php?title=Scrum_(desenvolvimento_de_software)&oldid=55061933)). *Wikipédia, a enciclopédia livre*. 7 de maio de 2019

Ver também

- [Ken Schwaber](#)
- [John Scumniotales](#)
- [Jeff Sutherland](#)

Ligações externas

- [Guia oficial do Scrum, em português \(PDF\) \(http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-Portuguese-BR.pdf\)](http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-Portuguese-BR.pdf)
- [Audio livro do Guia oficial do Scrum em Português \(https://www.youtube.com/watch?v=TMxjxtxZbBg\)](https://www.youtube.com/watch?v=TMxjxtxZbBg)
- [Comunidade Scrum no Brasil \(http://scrumbrasil.com.br/\)](http://scrumbrasil.com.br/)
- [Comunidade Scrum em Portugal \(http://www.scrumpt.com\)](http://www.scrumpt.com)
- «Uma introdução ao Scrum» (<http://www.mountaingoatsoftware.com/scrum-a-presentation>). (título em inglês: A Reusable Scrum Presentation)
- [Referências online do livro Scrum - Guia Prático para Projetos Ágeis, de Cesar](#)

Esta página foi editada pela última vez às 10h25min de 22 de julho de 2020.

Este texto é disponibilizado nos termos da licença Atribuição-CompartilhaIgual 3.0 Não Adaptada (CC BY-SA 3.0) da Creative Commons; pode estar sujeito a condições adicionais. Para mais detalhes, consulte as condições de utilização.

Obtida de "[https://pt.wikipedia.org/w/index.php?title=Scrum_\(desenvolvimento_de_software\)&oldid=58833680](https://pt.wikipedia.org/w/index.php?title=Scrum_(desenvolvimento_de_software)&oldid=58833680)"