Scrum

Scrum é uma metodologia ágil para gestão e planejamento de projetos de software. Se você chegou até aqui interessado em fazer uma das certificações disponíveis para Scrum, veja por que dizemos não à certificação?

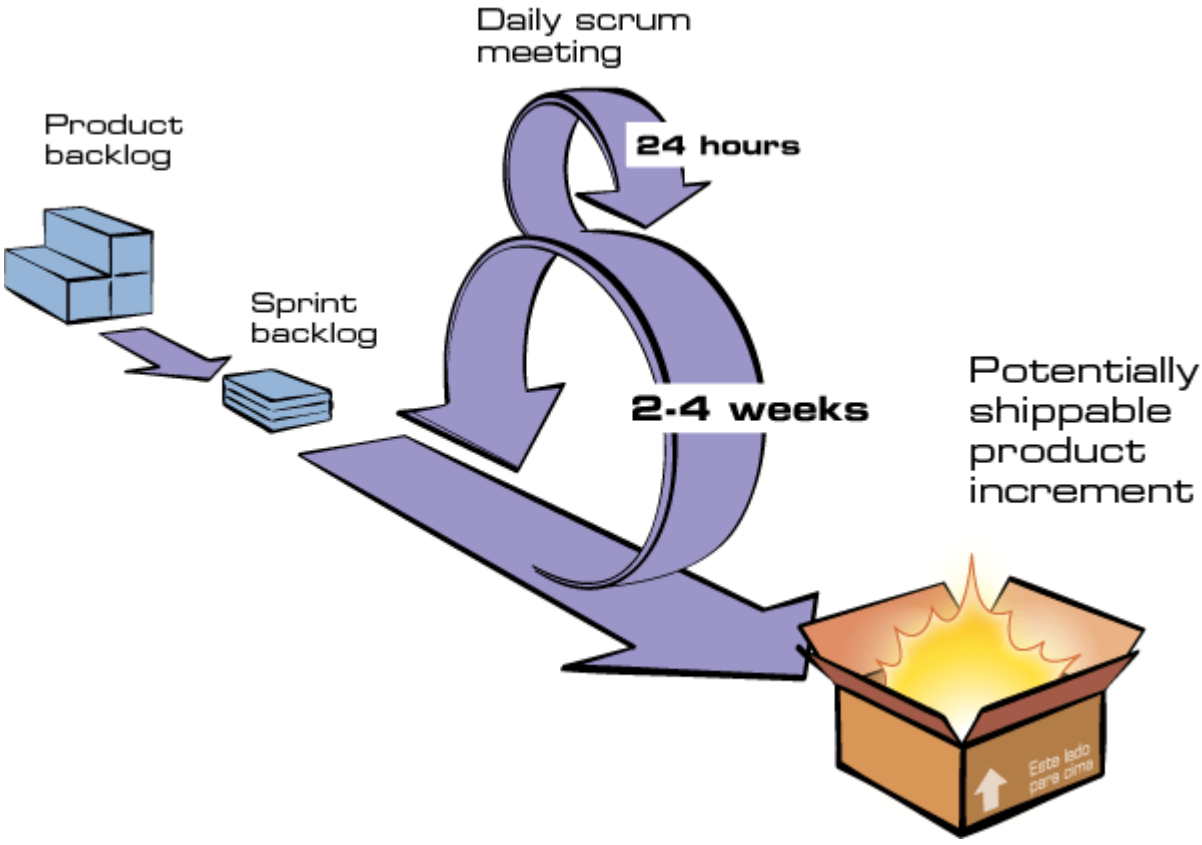
No Scrum, os projetos são divididos em ciclos (tipicamente mensais) chamados de **Sprints**. O **Sprint** representa um Time Box dentro do qual um conjunto de atividades deve ser executado. Metodologias ágeis de desenvolvimento de software são iterativas, ou seja, o trabalho é dividido em iterações, que são chamadas de Sprints no caso do Scrum.

As funcionalidades a serem implementadas em um projeto são mantidas em uma lista que é conhecida como Product Backlog. No início de cada Sprint, faz-se um Sprint Planning Meeting, ou seja, uma reunião de planejamento na qual o Product Owner prioriza os itens do Product Backlog e a equipe seleciona as atividades que ela será capaz de implementar durante o Sprint que se inicia.

As tarefas alocadas em um Sprint são transferidas do Product Backlog para o Sprint Backlog.

A cada dia de uma Sprint, a equipe faz uma breve reunião (normalmente de manhã), chamada Daily Scrum. O objetivo é disseminar conhecimento sobre o que foi feito no dia anterior, identificar impedimentos e priorizar o trabalho do dia que se inicia.

Ao final de um Sprint, a equipe apresenta as funcionalidades implementadas em uma Sprint Review Meeting. Finalmente, faz-se uma Sprint Retrospective e a equipe parte para o planejamento do próximo Sprint. Assim reinicia-se o ciclo. Veja a ilustração abaixo:



Product Backlog

O **Product Backlog** é uma lista contendo todas as funcionalidades desejadas para um produto. O conteúdo desta lista é definido pelo Product Owner. O Product Backlog não precisa estar completo no início de um projeto. Pode-se começar com tudo aquilo que é mais óbvio em um primeiro momento. Com o tempo, o Product Backlog cresce e muda à medida que se aprende mais sobre o produto e seus usuários.

Durante o Sprint Planning Meeting, o Product Owner prioriza os itens do Product Backlog e os descreve para a equipe. A equipe então determina que itens será capaz de completar durante a Sprint que está por começar. Tais itens são, então, transferidos do Product Backlog para o Sprint Backlog. Ao fazer isso, a equipe quebra cada item do Product Backlog em uma ou mais tarefas do Sprint Backlog. Isso ajuda a dividir o trabalho entre os membros da equipe. Podem fazer parte do Product Backlog tarefas técnicas ou atividades diretamente relacionadas às funcionalidades solicitadas.

Sprint Planning Meeting

O Sprint Planning Meeting é uma reunião na qual estão presentes o Product Owner, o Scrum Master e todo o Scrum Team, bem como qualquer pessoa interessada que esteja representando a gerência ou o cliente.

Durante o Sprint Planning Meeting, o Product Owner descreve as funcionalidades de maior prioridade para a equipe. A equipe faz perguntas durante a reunião de modo que seja capaz de quebrar as funcionalidades em tarefas técnicas, após a reunião. Essas tarefas irão dar origem ao Sprint Backlog.

O Product Owner não precisa descrever todos os itens que estão no Product Backlog. Dependendo do tamanho do Product Backlog e da velocidade da equipe, pode ser suficiente descrever apenas os itens de maior prioridade, deixando a discussão dos itens de menor prioridade para o próximo Sprint Planning Meeting.

Coletivamente, o Scrum Team e o Product Owner definem um objetivo para o Sprint, que é uma breve descrição daquilo que se tentará alcançar no Sprint. O sucesso do Sprint será avaliado mais adiante no Sprint Review Meeting em relação ao objetivo traçado para o Sprint.

Depois do Sprint Planning Meeting, a equipe Scrum se encontra separadamente para conversar sobre o que eles escutaram e decidir quanto eles podem se comprometer a fazer no Sprint que será iniciado. Em alguns casos, haverá negociação com o Product Owner, mas será sempre responsabilidade da equipe determinar o quanto ela será capaz de se comprometer a fazer.

Sprint Backlog

O Sprint Backlog é uma lista de tarefas que o Scrum Team se compromete a fazer em um Sprint. Os itens do Sprint Backlog são extraídos do Product Backlog, pela equipe, com base nas prioridades definidas pelo Product Owner e a percepção da equipe sobre o tempo que será necessário para completar as várias funcionalidades.

Cabe a equipe determinar a quantidade de itens do Product Backlog que serão trazidos para o Sprint Backlog, já que é ela quem irá se comprometer a implementá-los.

Durante um Sprint, o Scrum Master mantém o Sprint Backlog atualizando-o para refletir que

tarefas são completadas e quanto tempo a equipe acredita que será necessário para completar aquelas que ainda não estão prontas. A estimativa do trabalho que ainda resta a ser feito no Sprint é calculada diariamente e colocada em um gráfico, resultando em um Sprint Burndown Chart.

Product Owner

O Product Owner é a pessoa que define os itens que compõem o Product Backlog e os prioriza nas Sprint Planning Meetings.

O Scrum Team olha para o Product Backlog priorizado, seleciona os itens mais prioritários e se compromete a entregá-los ao final de um Sprint (iteração). Estes itens transformam-se no Sprint Backlog.

A equipe se compromete a executar um conjunto de atividades no Sprint e o Product Owner se compromete a não trazer novos requisitos para a equipe durante o Sprint. Requisitos podem mudar (e mudanças são encorajadas), mas apenas fora do Sprint. Uma vez que a equipe comece a trabalhar em um Sprint, ela permanece concentrada no objetivo traçado para o Sprint e novos requisitos não são aceitos.

Daily Scrum

A cada dia do Sprint a equipe faz uma reunião diária, chamada **Daily Scrum**. Ela tem como objetivo disseminar conhecimento sobre o que foi feito no dia anterior, identificar impedimentos e priorizar o trabalho a ser realizado no dia que se inicia.

Os Daily Scrums normalmente são realizadas no mesmo lugar, na mesma hora do dia. Idealmente são realizados na parte da manhã, para ajudar a estabelecer as prioridades do novo dia de trabalho.

Todos os membros da equipe devem participar do Daily Scrum. Outras pessoas também podem estar presentes, mas só poderão escutar. Isso torna os Daily Scrums uma excelente forma para uma equipe disseminar informações sobre o estado do projeto.

O Daily Scrum não deve ser usado como uma reunião para resolução de problemas. Questões levantadas devem ser levadas para fora da reunião e normalmente tratadas por um grupo menor de pessoas que tenham a ver diretamente com o problema ou possam contribuir para solucioná-lo. Durante o Daily Scrum, cada membro da equipe provê respostas para cada uma destas três perguntas:

- O que você fez ontem?
- O que você fará hoje?
- Há algum impedimento no seu caminho?

Concentrando-se no que cada pessoa fez ontem e no que ela irá fazer hoje, a equipe ganha uma excelente compreensão sobre que trabalho foi feito e que trabalho ainda precisa ser feito. O Daily Scrum não é uma reunião de status report na qual um chefe fica coletando informações sobre quem está atrasado. Ao invés disso, é uma reunião na qual membros da equipe assumem compromissos perante os demais.

Os impedimentos identificados no Daily Scrum devem ser tratados pelo Scrum Master o mais rapidamente possível.

Sprint Review Meeting

Ao final de cada Sprint é feito um Sprint Review Meeting. Durante esta reunião, o Scrum Team mostra o que foi alcançado durante o Sprint. Tipicamente, isso tem o formato de um demo das novas funcionalidades.

Os participantes do Sprint Review tipicamente incluem o Product Owner, o Scrum Team, o Scrum Master, gerência, clientes e engenheiros de outros projetos.

Durante o Sprint Review, o projeto é avaliado em relação aos objetivos do Sprint, determinados durante o Sprint Planning Meeting. Idealmente, a equipe completou cada um dos itens do Product Backlog trazidos para fazer parte do Sprint, mas o importante mesmo é que a equipe atinja o objetivo geral do Sprint.

Sprint Retrospective

O Sprint Retrospective ocorre ao final de um Sprint e serve para identificar o que funcionou bem, o que pode ser melhorado e que ações serão tomadas para melhorar.

Scrum Team

O Scrum Team é a equipe de desenvolvimento. Nela, não existe necessariamente uma divisão funcional através de papéis tradicionais, tais como programador, designer, analista de testes ou arquiteto. Todos no projeto trabalham juntos para completar o conjunto de trabalho com o qual se comprometeram conjuntamente para um Sprint.

Um Scrum Team típico tem de 6 a 10 pessoas, embora haja relatos de projetos Scrum com equipes maiores. A principal abordagem para trabalhar com equipes grandes no Scrum é usando o conceito de "Scrum of Scrums". Cada Scrum Team trabalha normalmente, mas cada equipe também contribui com uma pessoa que deverá freqüentar o Scrum of Scrums Meeting para coordenar o trabalho de múltiplas equipes Scrum. Esses encontros são análogos aos Daily Scrums, mas não acontecem necessariamente todos os dias. Fazer essa reunião duas ou três vezes por semana tende a ser suficiente na maioria das organizações.

Scrum Master

O Scrum Master procura assegurar que a equipe respeite e siga os valores e as práticas do Scrum. Ele também protege a equipe assegurando que ela não se comprometa excessivamente com relação àquilo que é capaz de realizar durante um Sprint.

O Scrum Master atua como facilitador do Daily Scrum e torna-se responsável por remover quaisquer obstáculos que sejam levantados pela equipe durante essas reuniões.

O papel de Scrum Master é tipicamente exercido por um gerente de projeto ou um líder técnico, mas em princípio pode ser qualquer pessoa da equipe.

Reunião em Pé

Stand up meeting é uma breve reunião realizada diariamente, normalmente de manhã, pela equipe de desenvolvimento com o objetivo de compartilhar informações sobre o projeto e priorizar suas atividades.

Trata-se de um diálogo entre todos os membros da equipe, se possível envolvendo também a presença do cliente. Em qualquer diálogo presencial, existem pelo menos duas coisas que são compartilhadas de maneira bastante rica: informações e emoções.

Informações

Em projetos XP, cada membro da equipe tem acesso a toda e qualquer parte do código e, acima de tudo, tem o direito de alterar qualquer parte do sistema, a qualquer momento, sem ter que pedir permissão a ninguém. Essa é uma prática conhecida como Código Coletivo. A parte boa dessa prática é que o desenvolvedor consegue avançar muito rapidamente, pois nunca fica dependendo de outra pessoa, ou da autorização de alguém, para editar uma parte do código. Por outro lado, isso exige muita troca de informações, visto que tudo o que está acontecendo, em qualquer parte do sistema, interessa a todos os membros da equipe.

O diálogo diário, realizado no stand up meeting, permite que cada desenvolvedor descreva brevemente o que fez no dia anterior, eventuais problemas que detectou, soluções interessantes que foram criadas etc. Não é necessário entrar em detalhes. A idéia é que as pessoas saibam o que está acontecendo e quem fez o que. Se um desenvolvedor se interessar bastante por um assunto a respeito do qual outra pessoa trabalhou no dia anterior, ele pode, após o stand up meeting, se reunir com essa pessoa e discutir a solução com maior nível de detalhe. Pode até resolver fazer par com aquela pessoa durante o dia de trabalho que está começando.

À primeira vista, o stand up meeting parece ser demorado. Afinal, se cada desenvolvedor tiver que explicar tudo o que fez no dia anterior, certamente isso pode consumir muito tempo. Entretanto, a idéia é fazê-lo diariamente e há várias razões importantes para ser assim. A primeira é que um dia de trabalho é um período relativamente curto. Descrever o que aconteceu em um dia é diferente de descrever os acontecimentos de uma semana, um mês ou um ano. Um dia é muito pouco tempo e normalmente não se produz uma quantidade enorme de coisas interessantes em um único dia. Sendo assim, existe a tendência de que cada desenvolvedor tenha pouco o que dizer se o stand up meeting realmente for conduzido com frequência diária. Apesar de pouco, o que ele tiver a dizer interessa a todos, porque o código é coletivo. De tempos em tempos, haverá alguma coisa realmente muito significativa que tenha sido feita por um par no dia anterior. Nestes casos, o stand up meeting pode acabar tomando mais tempo, porém o valor deste tempo é mais alto devido à utilidade da informação que se está transmitindo.

O stand up meeting é uma prática regida pelo valor da comunicação. Entretanto, é importante notar como também incorpora bem o valor do feedback. A cada dia de trabalho, podemos acertar em diversos pontos do projeto, mas também podemos errar. Equipes XP não esperam ser perfeitas e sabem que erram. Entretanto, isso não é temido, pois trata-se de um aspecto básico do ser humano: errar. O que realmente tememos é descobrir que erramos tarde demais, porque normalmente o custo de corrigir um erro cresce bastante quanto mais tempo levamos para detectá-lo e corrigi-lo. O stand up meeting é uma oportunidade para que os membros da equipe detectem e discutam problemas que tenha surgido no dia anterior, de modo que se possa priorizar ou não a correção dos mesmos e, de modo que se possam compartilhar sugestões sobre como tratá-los.

Quando os problemas são apresentados a todos os desenvolvedores, é comum que surjam várias sugestões e, dessa variedade, surge freqüentemente uma forma rápida e simples de solucionar a questão.

Se alguma coisa muito grave é detectada no stand up meeting, o gerente do projeto e o próprio cliente ficam sabendo rapidamente. Pois qualquer problema terá sido detectado há no máximo um dia de trabalho. Assim, o gerente, o cliente e os desenvolvedores podem criar soluções enquanto esse problema ainda não deu origem a outros.

Essa reunião matinal também é uma demonstração da aplicação do valor da simplicidade. Existem muitas formas de se transmitir informações, mas convenhamos, quando as pessoas estão próximas umas das outras, nada é mais simples do que conversar. Usar ferramentas é um caminho, mas isso normalmente envolve aprender a usá-las e a conviver com seus problemas. Além disso, não é necessário passar muito tempo ligado a computadores para saber que eles falham, os sistemas operacionais falham, travam, um dia funcionam e o no outro não e assim perde-se tempo, quando existem soluções mais simples e menos propensas a erros.

O stand up meeting também é um mecanismo para expressar e treinar a coragem dos membros da equipe. Em muitas empresas, os desenvolvedores escondem problemas potenciais com medo de sofrerem uma punição do chefe, do cliente ou de qualquer pessoa com poder na organização. De fato, muitas organizações são regidas pela cultura do medo. O stand up meeting é um espaço aberto onde as pessoas são incentivadas a falar tudo o que está acontecendo e são valorizadas por fazer isso. Trata-se de um aspecto importante porque com a prática, os membros da equipe perdem o medo de revelar os problemas à medida em que percebem que são valorizados por fazer isso. Além disso, os desenvolvedores passam a se expressar mais e com frequência, o que normalmente é ótimo para treinar a habilidade de comunicação. A mesma de que ele irá necessitar, por exemplo, para se comunicar bem com o cliente. Da mesma forma que é necessário coragem para dizer o que realmente está acontecendo, o stand up meeting acaba servindo como uma forma de treinar e desenvolver essa coragem diariamente.

Finalmente, equipes XP costumam utilizar radiadores de informações como o Quadro de Acompanhamento Diário. Trata-se de uma tabela, desenhada em um quadro branco, contendo informações sobre todas as histórias da iteração. Nela, colocam-se as estimativas de cada história, quanto tempo foi gasto em cada uma por dia da iteração, que tarefas extras foram executadas etc. É um quadro importante porque gera muita visibilidade para todos os membros da equipe, incluindo o cliente. Com esse quadro, não há necessidade de perguntar nada para o gerente para saber o estado do projeto em um iteração. Basta olhar para a parede. As informações estão lá. Projetos XP atualizam esse quadro diariamente, sempre nos stand up meetings. Assim, nenhum desenvolvedor precisa preencher uma folhinha dizendo quantas horas trabalhou por dia, por exemplo. Ao invés disso, ele faz algumas marcações simples em um quadro, uma vez por dia, durante o stand up meeting. Todos os desenvolvedores fazem isso durante essa reunião e leva apenas alguns segundos para cada pessoa marcar a sua informação. Quando todos o fazem em conjunto, essa prática tende a se manter de forma mais consistente do que quando realizamos uma atividade dessas sozinhos. Além disso, é um momento de socilização do qual todos participam.

Emoções

É possível transmitir informações através de ferramentas. Porém, emoções também informam e estas são muito difíceis de serem compartilhadas através de ferramentas. Durante um stand up meeting, os membros da equipe conseguem avaliar e sentir como está o clima da equipe. Olhando para cada pessoa, é possível observar se estão satisfeitas com o trabalho que está em andamento, dá para ver se estão cansadas ou energizadas, se estão entediadas ou empolgadas, se estão de acordo com os rumos do projeto etc. Muitas informações importantes jamais são verbalizadas, mas frequentemente basta você olhar para a expressão facial de uma pessoa para saber que ela discorda plenamente de alguma coisa ou para notar que ela está cansada, ou aborrecida. Nestes momentos, um bom coach (bem como qualquer membro da equipe) deve atuar pedindo à

pessoa que expresse seus sentimentos. Muitas vezes vivenciamos grandes viradas em projetos, viradas extremamente positivas, porque alguém notou um problema na expressão de um dos desenvolvedores.

Há pouco tempo, por exemplo, tínhamos iniciado um projeto em um dos nossos clientes e, em uma reunião dessas, um membro da equipe demonstrava pela expressão facial que não estava muito satisfeito com o rumo das coisas. Ele não disse isso, apenas pareceu insatisfeito pela expressão que fazia. Então perguntei o que o estava incomodando. Ele me explicou que tinha a sensação de que aquele projeto não deveria produzir um software para web, como estava sendo feito. Na opinião dele, pelas características do projeto, faria mais sentido que fosse uma aplicação desktop. Discutimos isso com todos e ficou bastante claro que ele estava coberto de razão. Então, decidimos reverter o que estávamos fazendo e refazer algumas coisas para direcionar o software para desktop.

Hoje, passados alguns meses, percebe-se que tomamos a decisão certa. Felizmente descobrimos isso bastante cedo devido à oportunidade de fazer um stand up meeting e interpretar as emoções que estavam no ar e não apenas as palavras verbalizadas. Se tivéssemos continuado no caminho original, talvez tivéssemos chegado à mesma conclusão em algum outro momento. Mas, certamente demoraria mais e o custo de reverter o projeto para desktop acabaria sendo maior. Portanto, emoções também representam um feedback importante que precisa ser levado em conta a todo o momento.

Para finalizar, precisamos compreender que o stand up meeting é uma forma de priorizar o que será feito em cada dia de trabalho. Nós priorizamos para assegurar que estamos fazendo o que é mais importante a cada momento do projeto e isso é feito porque o maior objetivo do XP é entregar um fluxo constante de valor para o cliente. Isso só é possível quando planejamos cuidadosamente o que será feito, isto é, quando escolhemos a cada dia fazer o que mais tenha potencial de gerar valor naquele momento. Sem essa priorização diária, é fácil um desenvolvedor acabar gastando tempo com algo que pode até ser útil, mas não era a coisa mais importante naquele ponto do projeto.

Em princípio, é fácil decidir o que deve ser feito a cada dia de trabalho. Basta olhar para o mural, onde a equipe coloca os cartões contendo as histórias da iteração, e verificar que histórias ainda precisam ser finalizadas. Entretanto, às vezes surgem problemas inesperados no dia anterior. Nesses casos, a equipe usa o stand up meeting para decidir se esses problemas devem ser tratados já no dia que está sendo iniciado, se devem ser deixados para uma iteração futura etc.

A priorização em equipe só pode ser feita de forma adequada quando temos a noção do todo, isto é, quando sabemos o que está acontecendo em cada parte do projeto, com cada pessoa da equipe e não apenas conosco. Pois, algo que eu possa considerar muito sério e prioritário no momento, pode ser absolutamente irrelevante quando somos informados de um problema bem mais significativo que está acontecendo em outra parte do projeto. Nos dias em que isso acontece, talvez faça mais sentido deixar o meu problema de lado e ir ajudar outra pessoa a resolver o problema mais sério que está afetando toda a equipe. Sem stand up meeting diário, é mais difícl notar e atuar rapidamente sobre situações como essas.