

02

Product Backlog e Sprint Backlog

Transcrição

Sabendo o que são **história** e **tarefa**, passaremos para as listas priorizadas e ordenadas dos itens. Elas auxiliam na organização e fazem o P.O. manter tudo em ordem, facilitando também a visualização da importância relativa de cada item.

O artefato padrão no Scrum é o **Product Backlog**, uma lista ordenada, que prioriza todos os itens de negócio e histórias até o final do projeto.

Quem cuida dessa lista é quem tem o domínio sobre o que entra e sai, e essa pessoa, o *Product Owner*, também estabelece prioridades nessa lista. O P.O. é o dono - *owner* - e manda nela, mas o cliente, os desenvolvedores e o Scrum Master podem influenciar o P.O. através de sugestões de re-priorização, colocando um item a mais ou tirando outro... Todo mundo pode dar sua opinião! Mas quem decide é o *Product Owner*.

Resumindo, a lista é do P.O., e nela constam os itens que devem ser desenvolvidos no projeto como um todo. Os itens mais prioritários são aqueles que estão no topo da lista e provavelmente já pensamos melhor a respeito deles quebrando-os em funcionalidades menores. Fica até mais fácil estimá-los e levá-los para o *planning*.

O processo de quebrar histórias muito grandes em menores de maneira que cada uma siga agregando valor ao usuário final, é um processo que chamamos de **refinamento** - o *grooming* - do Product Backlog.

Os itens menos prioritários, mais abaixo na lista de prioridade, não precisam ser todos refinados. É até bom que eles não estejam, pois não sabemos se serão úteis para o projeto futuramente. Pode ser que apareça algo mais urgente ou que agregue maior valor ao projeto, e neste caso o P.O. pode desconsiderar um item no qual já investimos certa quantidade de trabalho de refinamento.

Se algo perde importância no Backlog o item acaba ficando para baixo na lista de prioridade, e isso vai sendo deixado de lado até que o projeto acabe. Se ainda for interessante realizar itens e tarefas que ficaram para fora do prazo estimado, pode ser feito um adendo de contrato, prosseguindo-se a execução desses itens caso agregarem valor suficiente.

Ao chegar no *planning*, o P.O. traz o topo do *Product Backlog*, e isso é discutido com o time e quebrado em histórias e tarefas, fazendo com que seja mais simples estimá-las. Ao quebrarmos as tarefas em histórias, fica ainda mais simples de compreendermos suas complexidades.

É bastante natural que no Sprint Backlog não existam só histórias, mas também tarefas. Isso inclusive é esperado. No *"Scrum Guide"*, comenta-se que cerca de 70% das tarefas vão surgir do *planning*.

O *Sprint Backlog*, diferentemente, é do time como um todo, e a regra é que a equipe pode renegociar prioridades (lembrando que o time é composto de desenvolvedores, P.O. e Scrum Master). Assim, é possível mudar o escopo de maneira a agregar valor - essa é uma decisão interna do time.

O Sprint Backlog possui a regra de que o cliente não pode dar sua opinião. Se algo entrou para ser feito nessa Sprint, então é porque são itens de maior prioridade. Se o cliente pensa em mexer nisso é porque o P.O. não conversou direito com ele, ou as prioridades não estão bem estabelecidas, e ele muda de opinião o tempo inteiro.

A única razão pela qual se faz um cancelamento de Sprint no Scrum é quando **o Sprint Backlog se altera tanto a ponto da meta não fazer mais sentido**. Os cancelamentos também podem ocorrer no caso de **problema de comunicação grave**,

um cliente muito indeciso ou eventualmente outro tipo de razão externa, como uma justificativa legal. No entanto tais situações não são comuns ou não devem ser recorrentes.

Retomando:

- O *Product Backlog* é a lista priorizada das histórias que agregam valor para o cliente. São histórias que envolvem o projeto inteiro. Somente o P.O. mexe nele, mas todo o time pode palpitar;
- O *Sprint Backlog* engloba histórias e tarefas que estão no topo das prioridades. Conforme os itens entram nele, já quebramos as histórias em tarefas. No Sprint Backlog o time altera essas tarefas sem que o cliente palpite sobre elas.