



TI

Como segmentar o trabalho ajudou um desenvolvedor a enxergar problemas

No segundo artigo de nossa série com a Theodo, o CTO fala sobre um experimento que mostra como a resolução de problemas é mais fácil quando o trabalho está claramente definido.

BENJAMIN GRANDFOND



Michael Ballé nos visitou recentemente na Theodo. Depois de sua caminhada pelo *gemba*, ele nos disse sem rodeios que não estávamos aplicando adequadamente a puxada, pois não entendíamos seu propósito.

Em nossa indústria, os sprints geralmente duram entre duas semanas e um mês, e os históricos de usuários, entre um e cinco dias. Então pensamos que estávamos indo muito bem

Ao vê-lo surpreso, tentamos explicar-lhe por que pensávamos estar realmente puxando. "Temos sprints que só duram uma semana, e nossos históricos de usuários duram menos de um dia", disse Benoît, nosso cofundador (um sprint é um período de tempo durante o qual produzimos recursos para o software dos clientes, organizados





em históricos de usuários). Em nossa indústria, os sprints geralmente duram entre duas semanas e um mês, e os históricos de usuários, entre um e cinco dias. Então pensamos que estávamos indo muito bem.

"Por favor! Na fabricação, se um operador leva mais de 10 minutos para fazer algo, significa que ele não está dominando o conteúdo do trabalho", respondeu Michael. Logo depois disso, Benoît, nosso coach lean Régis Medina e eu começamos a pensar em maneiras de quebrar o trabalho de nossos desenvolvedores em tarefas menores de 10 minutos de duração.

DIVIDA O TRABALHO, ENXERGUE OS PROBLEMAS

Escolhi experimentar com um dos líderes tecnológicos da equipe, Nicolas Boutin. No momento, Nicolas estava liderando uma equipe de dois desenvolvedores que haviam falhado com cinco sprints nas 10 semanas anteriores.

Passei duas horas com ele enquanto ele construía um histórico de usuário. Antes de começar a codificar, pedi que listasse e definisse cada etapa (de 10 minutos no máximo) que teria que executar para construir o histórico. Então, ele calculou o tempo para cada etapa. Mesmo se ele não soubesse fazer algo, pensasse que não seria capaz de terminar a etapa ou acabasse o tempo, eu disse que ele poderia me pedir ajuda sempre que precisasse - assim como um operador em uma planta de montagem da Toyota faria ao puxar o fio *andon*.

Nicolas estava trabalhando em um formulário de inscrição para abrir uma apólice de seguro de vida on-line, na qual o cliente adiciona a quantidade mínima de dinheiro que ele deseja adicionar. Nicolas dividiu o histórico do usuário em oito etapas, com um total de 22 minutos e 10 segundos (em vez das aproximadamente quatro horas, que a equipe inicialmente havia estimado):

1. Preencher os formulários on-line para chegar à etapa 4 e ver a página onde o resultado final seria visível - 3 minutos.
2. Encontrar um código semelhante para copiar e colar - 30 segundos.
3. Encontrar o lugar para colar o código copiado, colá-lo e adaptá-lo com dados de amostra - 10 segundos.
4. Escrever um caso de teste automático - 10 minutos.
5. Substituir os dados da amostra por dados reais obtidos do banco de dados de acordo com o contrato assinado - 1 minuto.
6. Testar no navegador - 30 segundos.
7. Publicar o código na plataforma de validação - 2 minutos.
8. Testar na plataforma de validação - 5 minutos.





Como empresas líderes estão conduzindo seu processo de inovação – [Veja as sessões do Lean Summit 2018](#)

Se não encontrasse nenhum problema, Nicolas poderia entregar o histórico 10 vezes mais rápido do que o esperado anteriormente.

Observei com o cronômetro na mão quando ele começou a codificar e escrever tudo o que ele fazia. O primeiro problema surgiu na segunda etapa: Nicolas lutou para encontrar as três linhas de código de que precisava no arquivo de mais de 300 linhas. O arquivo era obviamente muito longo, e demorou dois minutos para encontrar o código certo.

Depois que ele colou as linhas no arquivo certo, a página caiu. O design não era como o esperado, com alguns blocos de conteúdo desalinhados ou ausentes. Percebemos que ele não havia copiado todas as linhas de que precisava. Duas horas depois, Nicolas completou seis das oito etapas que ele havia listado. Três etapas foram realizadas no tempo estimado, e duas faltavam no fluxo. Ele puxou o *andon* quatro vezes. No final, levou pouco menos de quatro horas para completar o histórico do usuário, assim como a equipe havia estimado (portanto, o longo processo não teve impacto no plano de produção).

Tiramos um tempo para analisar os problemas que Nicolas encontrou em cada etapa. Isso incluiu: falta de conhecimento da tecnologia e das ferramentas que ele usou; dificuldade para ler o código e falta de adesão às convenções de codificação interna; o fato de que seu IDE (Integrated Development Environment - basicamente o editor de código) não foi configurado.

Sempre que possível, abordamos a causa raiz do problema para solucioná-lo imediatamente.

Sempre que possível, abordamos a causa raiz do problema para solucioná-lo imediatamente. Para todos os problemas que não puderam ser resolvidos imediatamente, Nicolas adicionou uma tarefa a sua lista. No final do dia, ele estava entusiasmado. "Hoje é o dia que mais aprendi desde que comecei a trabalhar na Theodo", ele me disse.





SOLUCIONE PEQUENOS PROBLEMAS PARA CRIAR LÍDERES

O experimento foi tão bem sucedido que decidimos estendê-lo a toda a equipe que Nicolas lidera. Os resultados foram ótimos: por oito semanas seguidas, cada sprint foi entregue a tempo. Ao longo das últimas quatro semanas, eles até melhoraram sua produtividade em 40%. Além disso, à medida que resolvemos os problemas, fomos capazes de criar um espírito de *kaizen* na equipe e descobrimos que recebemos mais e mais oportunidades de desenvolvimento de liderança.

Hoje, cada desenvolvedor aborda um dos problemas mais importantes que encontra todos os dias

Hoje, cada desenvolvedor aborda um dos problemas mais importantes que encontra todos os dias. Antes, os problemas não eram claramente definidos - "Estamos meio dia atrasados" - o que significava que encontrar as causas iniciais era como procurar uma agulha no palheiro. Agora, eles investigam problemas em menos de 10 minutos depois deles ocorrerem, para que possam ser consertados imediatamente.

Como líder de equipe, ele ajuda cada desenvolvedor a crescer e a ter sucesso em seu trabalho

Por fim, achamos que o papel de Nicolas na equipe está muito mais claro. Como líder de equipe, ele ajuda cada desenvolvedor a crescer e a ter sucesso em seu trabalho. O feedback da equipe foi tão positivo que decidimos lançar essa iniciativa em outras equipes. Talvez uma ótima oportunidade para outro artigo?

Conheça o caminho de empresas líderes na sua transformação digital - [Participe do Lean Summit 2018](#)

Benjamin Grandfond é diretor de tecnologia da Theodo.

Fonte: [Planet Lean](#)

Publicado em 25/04/2018

