





Fundamentos Scrum

FUNDAMENTOS DE SCRUM



Conteúdo

Sobre o curso.....	4
Propósito	4
Público alvo	6
Tópicos abordados	6
Requisitos Ágeis	6
O Framework Scrum	7
Técnicas de Estimativa de Software.....	7
Planejamento com Scrum.....	7
Fique saudável com o Scrum.....	7
Introdução.....	8
Definindo Requisitos Ágeis.....	9
Requisitos e transferência de conhecimento.....	9
Especificação de Requisitos (do modo formal).....	11
Resultado esperado	11
Possíveis Interpretações	12
Especificação de Requisitos (formato Scrum)	12
Transparência	13
Inspeção	13
Adaptação.....	13
O Que é Scrum?	14
Teoria do Scrum.....	15

Funções Scrum	17
Equipe Scrum	17
O Scrum Master	17
Scrum Product Owner.....	18
Artefatos Scrum.....	19
Backlog do Produto	19
User Stories (Estórias de Usuário).....	19
Sprint Backlog	20
Técnicas de Estimativa.....	22
Estimativas de Esforço.....	22
Planning Poker® / Poker Scrum	22
Gráfico Burndown	24
Gráfico simples de Burndown	25
Definição de Pronto	26
Planejamento	27
Sprint	27
O Processo Sprint	28
Scrum saudável	33
Revisão	34
O Que é Scrum?.....	34
Atividades.....	39
Planning Poker    	41
O Baralho	41
O Processo.....	42

Avaliação.....	44
Referencias.....	51

Scrum Foundations

Fundamentos de Scrum

Sobre o curso

Times Scrum atingem melhores resultados com uma base de conhecimento sólida e este curso de 2 dias prepara os estudantes para iniciar o uso efetivo do Scrum imediatamente. O framework Scrum, assim como a sua mecânica e regras são abordados com ênfase em aplicações práticas.

Este curso é a introdução perfeita ao Scrum para qualquer pessoa que está iniciando no framework. Se o seu time necessita aprender o uso correto do Scrum ou simplesmente deseja reiniciar o esforço de uma implantação, este curso é indicado para você. Ele também é a base de conhecimento para outros treinamentos mais avançados como o Product Owner e Professional Scrum Master.

Propósito

Professional Scrum Foundations (PSF) é um treinamento rigoroso de 2 dias oferecido para solidificar os princípios fundamentais do Scrum para indivíduos e times. Erros comuns durante a adoção do Scrum têm sido cada vez mais vistos nas indústrias, empresas e times. Ao invés de treinar novamente esses times que vieram de uma implantação pobre do Scrum, o PSF expõe esses erros comuns, aumenta a consciência dos sintomas associados e fornece um guia prescritivo para evitar esses erros.

As experiências práticas oferecidas pelo curso preparam os alunos para iniciarem no Scrum, para sustentar hábitos que os levarão ao sucesso e, principalmente, para evitar as

“pegadinhas” comuns do Scrum. Os alunos aprenderão o básico do Scrum, como implantá-lo de forma efetiva e a manter seus times praticando comportamentos saudáveis.

Times que implementam Scrum sem o entendimento dos fundamentos do framework estão muito menos propensos a terem sucesso do que aqueles times que já experimentaram o Scrum em um ambiente de aprendizagem. Esse curso intencionalmente aborda os desafios comuns que os times enfrentarão durante a sua adoção, preparando os alunos para mitigar problemas que podem fazer a sua implantação falhar, incluindo:

- Timeboxes flexíveis
- Atraso na correção de bugs
- Definição de Pronto (DP) subjetiva
- Medir esforço ao invés de entregas
- Tendência subconsciente a voltar para o Waterfall
- Retrospectivas improdutivas
- Reuniões diárias não eficientes
- Backlog estruturados de maneira pobre
- Requisitos ambíguos ou ineficazes
- Membros do time não comprometidos
- Requisição de Trabalho sob demanda
- E muitos outros

Público alvo

O PSF é indicado para qualquer pessoa envolvida com um time Scrum. O curso é especialmente apropriado para empresas e times que estão se iniciando no Scrum ou estão tendo dificuldades com a sua implantação.

O curso de PSF não é indicado para indivíduos ou times que já utilizam o Scrum no seu dia a dia e estão procurando ajuda com tópicos mais avançados como “Escalando o Scrum” ou implantando um framework de ROI (Retorno sobre o investimento). Para essas necessidades e outros assuntos mais avançados o curso de Professional Scrum Master é mais indicado.

Tópicos abordados

Além dos assuntos abaixo, os alunos participarão ativamente de diversos exercícios reais e práticos para reforçar os pontos de aprendizagem. Praticar Scrum garante que os alunos concluirão o curso com novas habilidades além dos novos conhecimentos e teoria.

Requisitos Ágeis

Embora existam muitas técnicas efetivas para expressar requisitos, este módulo fornece aos times técnicas provadas para levantar, representar e decompor requisitos como itens de Backlog.

O Framework Scrum

Os alunos terão uma visão completa do framework de desenvolvimento de software Scrum e aprenderão sobre suas regras, princípios, artefatos e atividades.

Técnicas de Estimativa de Software

O time aprenderá a usar estimativas como uma ferramenta para o entendimento dos requisitos dentro do framework Scrum.

Essas técnicas formalizam o entendimento e elucidação dos requisitos e permitem que todo o time possa participar ativamente das atividades de planejamento do Backlog.

Planejamento com Scrum

Aprenda técnicas práticas de planejamento de releases, gerenciamento do Product Backlogs, como manter a Sprint dentro do planejado e a manter o comportamento do time saudável.

Fique saudável com o Scrum

Este módulo foca em erros comuns que os times cometem quando adotam Scrum e estratégias para evitá-los. Além disso, ele discute padrões e anti-padrões típicos, o que incentiva os alunos a praticarem uma conversa bastante corajosa levando a uma troca de experiências única entre alunos e instrutor.

Introdução

Jeff Sutherland e Ken Schwaber conceberam o processo Scrum no início dos anos 90. Em 1995, redigiram o código Scrum afim de apresentá-lo em uma conferência Oopsla em Austin, Texas (EUA) e publicaram o documento "Processo Scrum de Desenvolvimento de Software" (Scrum Software Development Process).

Ken e Jeff adotaram o nome 'Scrum' oriundo do artigo inovador 'O Novo Novo Jogo de Desenvolvimento de Produtos' (The New New Product Development Game) de Takeuchi e Nonaka, dois pensadores de gerenciamento muito experientes. Com o termo 'Scrum' Nonaka e Takeuchi se referiram ao jogo de Rugby para estressar a importância de equipe e algumas analogias entre equipes de esporte como Rugby e o sucesso no jogo de desenvolvimento de um novo produto. A pesquisa descrita em seu artigo mostrou como que um desempenho excepcional em desenvolvimento de um produto novo e complexo é alcançado quando equipes, em unidades auto organizadas de pessoas, recebem objetivos e não tarefas. As melhores equipes são aquelas que recebem direção e nela tem espaço para criarem suas próprias táticas de como alcançar seus objetivos em conjunto. Equipes requerem autonomia para alcançarem excelência. [h]

Perdendo no revezamento...

O estilo de "corrida de revezamento" aplicado ao desenvolvimento de produtos pode conflitar com os objetivos de velocidade e flexibilidade máximas. Ao invés disto, um estilo holístico, onde a equipe busca, como em um jogo de futebol, de forma integrada, chegar ao gol, com passes de bola, pode servir melhor às atuais necessidades competitivas.

Adaptado de "The New New Product Development Game", Hirotaka Takeuchi e Ikujiro Nonaka, Harvard Business Review, January 1986.

Definindo Requisitos Ágeis

Problema com requisitos de software no padrão IEEE-830:

- Trabalhosos para serem lidos, resultando em leituras superficiais ou sessões ignoradas;
- Múltiplas interpretações possíveis, propensa a erros;
- Presume que tudo é conhecido com antecedência;
- Clientes entendem a finalidade do produto final, mas, na maioria dos casos, ainda não consideraram todas as funcionalidades necessárias do produto.

Requisitos e transferência de conhecimento

A abordagem 'design primeiro' da metodologia em cascata requer que especialistas se encarreguem de esclarecer tarefas e requisitos, documentá-los, e então transmiti-los para aqueles que irão completá-los.

Tipo de Conhecimento	Descrição	Ágil	Tradicional
Conhecimento Tácito	Experiência profundamente pessoal, aptidões, percepções, introspecções, e know-how que se aplicam ou são indicados, mas não são expressados – residem em indivíduos e equipes.	Sim	Não
Conhecimento Explícito	Conhecimento que é codificado e transmitido aos outros através de diálogo, demonstrações, ou mídias tipo livros, desenhos e documentos.	Não compulsório	Mandatário

Nos projetos onde a metodologia em cascata foca na documentação para transferir conhecimentos e requisitos, a prática percebe que muito desse conhecimento é experimental e, portanto, impossível de transferir aos outros dessa forma [b]. O único modo eficaz de expressar este conhecimento é na esfera social através de trabalho em equipe e colaboração [c]. Repensar 'como requisitos são feitos' ao levar essa questão em consideração é necessário porque diminui o risco de fracasso dos projetos.

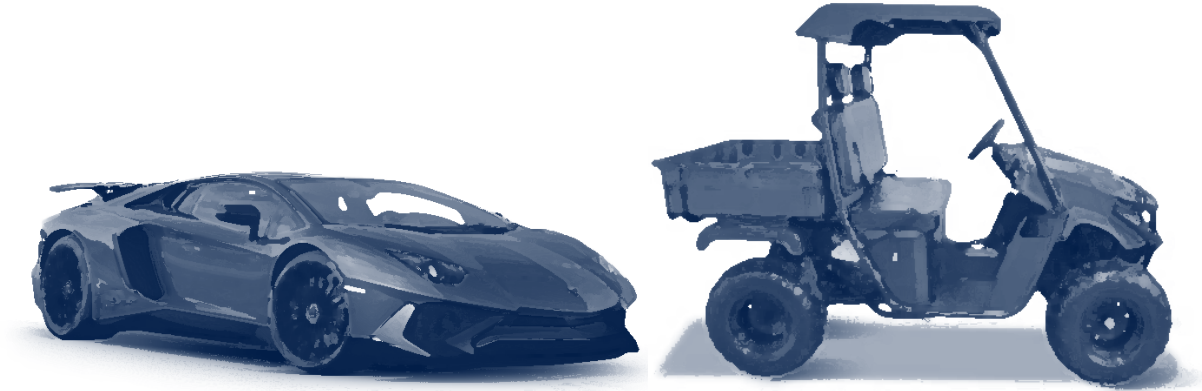
Especificação de Requisitos (do modo formal)

- O produto deve ter um motor de combustão;
- O produto deve ter quatro rodas;
- O produto de ter pneus de borracha montados em cada roda;
- O produto deve ter um volante;
- O produto deve ter carroceria resistente.

Resultado esperado



Possíveis Interpretações



Especificação de Requisitos (formato Scrum)

É recomendado definir requisitos focando na **MOTIVAÇÃO**, ao invés da implementação. Scrum é fundamentado nas teorias **empíricas** de controle de processo, ou empirismo. O empirismo afirma que o conhecimento vem da experiência e de tomada de decisões baseadas no que é conhecido. O Scrum emprega uma abordagem iterativa e incremental para aperfeiçoar a previsibilidade e o controle de riscos.

Estórias focam no que o usuário almeja, e são mais fáceis de serem interpretadas em requisitos na hora de definir as tarefas de desenvolvimento.

A vantagem de estórias de usuário são que focam exatamente no que o usuário precisa/quer sem entrar em detalhes de como fazê-lo.

Há diferentes recomendações de como definir estórias de usuário. Segue um modelo:

Como um [ator], Eu [quero/preciso] [ação] para que [conquista]

Ou, numa versão mais curta:

Como um [ator], Eu [quero/preciso] [conquista]

Três pilares apoiam a implementação de controle de processo empírico: transparência, inspeção e adaptação.

Transparência

Aspectos significativos do processo devem estar visíveis aos responsáveis pelos resultados. Esta transparência requer aspectos definidos por um padrão comum para os observadores compartilharem um mesmo entendimento do que está sendo visto.

Por exemplo:

- Uma linguagem comum referindo-se ao processo deve ser compartilhada por todos os participantes; e,
- Uma definição comum de "Pronto" deve ser compartilhada por aqueles que realizam o trabalho e por aqueles que aceitam o resultado do trabalho.

Inspeção

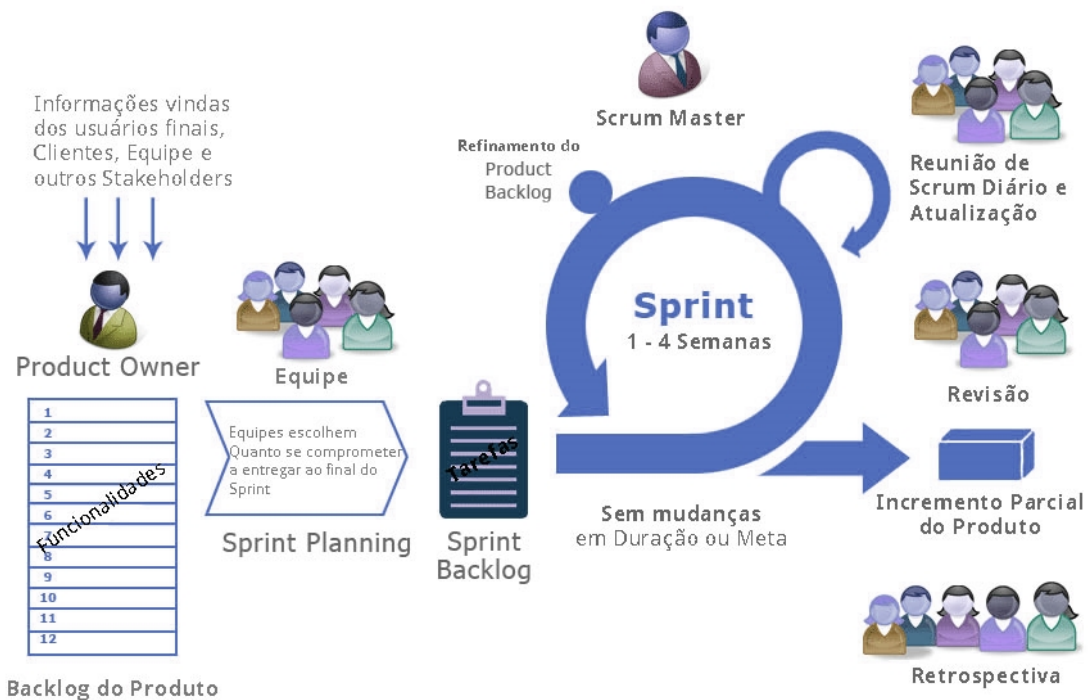
Os usuários Scrum devem, frequentemente, inspecionar os artefatos Scrum e o progresso em direção a detectar variações. Esta inspeção não deve, no entanto, ser tão frequente que atrapalhe a própria execução das tarefas. As inspeções são mais benéficas quando realizadas de forma diligente por inspetores especializados no trabalho a se verificar.

Adaptação

Se um inspetor determina que um ou mais aspectos de um processo desviou para fora dos limites aceitáveis, e que o produto resultado será inaceitável, o processo ou o material sendo produzido deve ser ajustado. O ajuste deve ser realizado o mais breve possível para minimizar mais desvios.

O Que é Scrum?

Scrum é uma leve estrutura ágil de gerenciamento de projetos usada principalmente para desenvolvimento de software. O mesmo descreve uma abordagem iterativa e incremental para o trabalho do projeto.



Pode ser usado em todo tipo de desenvolvimento de software: pacotes completos de software, partes de sistemas maiores, projetos internos e de clientes.

Teoria do Scrum

A estrutura Scrum implementa os pontos mais importantes definidos no manifesto Ágil [e]:

- Indivíduos e iterações ao invés de processos e ferramenta;
- Software que funciona ao invés de documentação extensiva;
- Colaboração com o cliente ao invés de negociações de contrato;
- Responder às mudanças ao invés de seguir o plano.

A estrutura de Scrum é bastante simples. Define somente algumas guias gerais com somente algumas regras, funções, artefatos e eventos. Apesar disso, cada um desses componentes é importante, serve a um propósito específico e essencial para o sucesso de seu uso. [f]

Os componentes principais da estrutura Scrum são:

- As três funções: Product Owner, Scrum Master e Equipe Scrum;
- Um Backlog priorizado que contenha todos os requisitos dos usuários finais e de Sprints;
- Eventos Scrum: Sprint Planning, Scrum Diário, Sprint Review e Retrospectiva da Sprint.

Importante em todos os projetos Scrum são a auto-organização e comunicação entre os integrantes da equipe. Não existe mais um gerente de projetos no sentido clássico. A estrutura Scrum de Scrum Master e Product Owner compartilham esta responsabilidade. No entanto, no final é a equipe que decide o quê e quanto trabalho poderá ser entregue em cada iteração (Sprint).

Outro aspecto central dentro da estrutura Scrum é a melhoria contínua: inspecionar e adaptar. As equipes Scrum têm que inspecionar e avaliar frequentemente os artefatos criados e seus processos para que possam adaptá-los e otimizá-los. Em meio a essas inspeções e adaptações a equipe irá otimizar os resultados, aumentar a previsibilidade e minimizar o risco geral do projeto.

A estrutura Scrum almeja lidar com o fato de que os requisitos provavelmente irão mudar rapidamente ou talvez não sejam conhecidos totalmente no início do projeto. Os requisitos mais detalhados são somente definidos quando estão prestes a serem implementados. Em Scrum, mudanças e otimizações do produtos, requisitos e processos são parte integral do ciclo de engenharia.

Outra parte fundamental da estrutura Scrum é a comunicação. O Product Owner trabalha de perto com a equipe Scrum para identificar e priorizar funcionalidades. Essas funcionalidades são escritas em user stories e armazenadas no Backlog do Produto. O Backlog do produto consiste em tudo que precisa ser feito para que o sistema de software seja entregue funcionando.

A equipe Scrum tem autoridade para selecionar somente as user stories que tem certeza que podem terminar durante as 2-4 semanas da Sprint. É permitido à equipe Scrum que se comprometa com suas próprias metas e trabalho que poderão completar com o melhor desempenho possível.

O Scrum Master é outra função importante da estrutura Scrum pois funciona como um mestre-servo junto à equipe Scrum. As tarefas deste são para garantir que a equipe Scrum compreenda como se opera Scrum, para proteger a equipe Scrum de interrupções externas, e para remover impedimentos que atrasem a equipe em alcançar sua produtividade máxima.

A estrutura Scrum é melhor usada em projetos menores, com somente uma equipe. No entanto, a introdução de mais funções como "Product Owner Chefe" funcionam em equipes maiores/múltiplas ou em projetos distribuídos.

Funções Scrum

Dentro da Estrutura Scrum temos as seguintes funções:

- A equipe Scrum;
- O Scrum Master;
- O Product Owner Scrum (ou PO).

Cada uma dessas funções tem um número de responsabilidades definidas, interação de perto com os outros integrantes e contribuem para o sucesso do projeto somente se cumprirem suas responsabilidades.

Equipe Scrum

Na estrutura Scrum todo o trabalho entregue ao cliente é feito por equipes Scrum. Uma equipe Scrum é um grupo de indivíduos trabalhando juntos para entregar os requisitos e funcionalidades em incrementos do produto.

Para trabalharem eficazmente é importante que cada membro da Equipe:

- Siga uma meta em comum;
- Siga as mesmas normas e regras;
- Demonstrem respeito uns com os outros.

O Scrum Master

Em geral o trabalho do Scrum Master é garantir que a equipe Scrum siga a metodologia Scrum, suas práticas e regras.

O Scrum Master faz parte da equipe Scrum e atua como um líder-servo da equipe. Em princípio, este pode ser um trabalho de tempo integral forçando o Scrum Master a não contribuir diretamente com os resultados da Sprint. Com o passar do tempo, após algumas Sprints os processos irão se solidificar e o fardo do Scrum Master irá diminuir e o mesmo poderá contribuir diretamente para a Meta da Sprint.

Sendo crucial a confiança entre a equipe e o Scrum Master, é ideal que a equipe escolha o Scrum Master. No entanto, é muito mais comum que a gerência escolha o mesmo. Para que o Scrum Master consiga a confiança do resto da equipe é importante que o mesmo não tenha nenhuma superioridade autoritária sobre o resto da equipe. De outra forma, a comunicação aberta e decisões terão deficiência.

Scrum Product Owner

O Product Owner (ou PO) tem função central dentro da estrutura Scrum. A maioria das responsabilidades do gerente de produtos clássico e do gerente de projetos se combinam em uma função única.

Ele representa o usuário final e/ou os Stakeholders, e é responsável por maximizar o valor do produto ao garantir que o trabalho correto é feito e no tempo certo.

Como consequência, o Product Owner tem que trabalhar muito próximo à Equipe Scrum e coordena suas atividades por toda vida do projeto. A ninguém mais é permitido indicar as prioridades de trabalho para a equipe Scrum.

O Product Owner tem várias responsabilidades:

- Gerenciar o Product Backlog (Backlog do Produto);
- Gerenciar os lançamentos ou releases;
- Gerenciar os Stakeholders (envolvidos/interessados);
- Trabalhar de perto à Equipe Scrum.

Claro que o Product Owner pode delegar certas atividades (como fisicamente manter o Backlog do Produto), mas no final ele continua o responsável.

Artefatos Scrum

Backlog do Produto

A definição mais simples do Backlog do produto é uma lista com tudo que o projeto precisa, sendo uma lista viva que pode crescer ou diminuir durante a vida útil do projeto. Essa lista substitui os artefatos tradicionais de especificação e requisitos. Esses itens podem ter certa natureza técnica e podem até serem centralizadas no usuário final, na forma de histórias de usuário (user stories). O dono do Backlog do Produto é o Product Owner. O Scrum Master, a equipe Scrum e outros Stakeholders podem contribuir para definir uma lista completa de tarefas para os itens do Backlog do Produto.

Outros artefatos podem ser usados pela equipe Scrum para facilitar a compreensão dos itens do Backlog do Produto, no entanto, esses artefatos não substituem o Backlog do Produto, seus detalhes ou conteúdo.

O Product Owner usa o Backlog do Produto durante a reunião de planejamento da Sprint para descrever seus itens para a equipe. Então, a equipe Scrum determina quais itens podem completar durante a próxima Sprint.

User Stories (Estórias de Usuário)

Os itens do Backlog do Produto são geralmente escritos na forma de Estórias de Usuário. Uma Estória conta brevemente sobre alguém usando o produto. Contém um nome, uma breve narrativa, um critério de aceite e condições para que a estória seja completada. A

vantagem de estórias de usuário são que focam exatamente no que o usuário precisa/quer sem entrar em detalhes de como fazê-lo.

Sprint Backlog

Dentro do Sprint Backlog estão todas as atividades requeridas para completar os itens comprometidos e listados no Backlog do Produto. Todos os itens devem ser estimados na base de horas por pessoa para que o progresso e esforço pendentes sejam acompanhados.

O Sprint Backlog é um artefato vivo e deve ser atualizado diariamente. Se um membro da Equipe começar a trabalhar em alguma atividade, seu nome deve ser gravado nessa tarefa do Backlog. Novas atividades podem ser adicionadas ao Sprint Backlog durante a Sprint. Ao final do dia todos os esforços remanescentes devem ser atualizados e isso irá definir quanto trabalho ainda falta até que a Meta da Sprint seja alcançada. A "Definição de Pronto" (abaixo) é usada para decidir quais itens estão prontos ou não.

O Sprint Backlog pode ser mantido eletronicamente em uma planilha de Excel ou em cartões de um quadro. O último tem mais vantagens (transparência e fácil acesso) mas também adiciona certa complexidade se a Equipe Scrum estiver distribuída em múltiplos lugares físicos (assunto discutido posteriormente).

Existem muitas outras ferramentas disponíveis no mercado, inclusive ferramentas proprietárias e comerciais:

- Atlassian
- Apprenda
- Borland
- CA Technologies
- HP

- IBM
- Polarion
- RallyDev
- Serena
- Smart Bear
- Team Forge
- Tech Excel
- Thoughtworks
- Visual Studio ALM/DevOps
- Visual Studio Online

A figura abaixo mostra um exemplo de como tal quadro pode ser organizado. A estrutura deve se adaptar para refletir as necessidades do projeto.

Item do Backlog do Produto	Tarefas	Tarefas em Andamento	Tarefas Completadas
[Item do Backlog]	<ul style="list-style-type: none"> [Tarefa] [Tarefa] [Tarefa] 	<ul style="list-style-type: none"> [Tarefa em Andamento] 	<ul style="list-style-type: none"> [Tarefa Completada]
[Item do Backlog]	<ul style="list-style-type: none"> [Tarefa] [Tarefa] 	<ul style="list-style-type: none"> [Tarefa em Andamento] 	<ul style="list-style-type: none"> [Tarefa Completada]
[Item do Backlog]	<ul style="list-style-type: none"> [Tarefa] [Tarefa] [Tarefa] 		

Técnicas de Estimativa

Estimativas de Esforço

Todos os itens do Backlog do produto têm uma estimativa que permite ao Product Owner priorizar os itens e planejar os lançamentos. Isso significa que o Product Owner necessita de avaliações honestas sobre a dificuldade do trabalho a ser feito. Apesar disso, é recomendado que o Product Owner não participe da estimativa para evitar pressionar (intencionalmente ou não) a Equipe Scrum.

A estrutura Scrum não prescreve um modo único de estimar o esforço. No entanto, dentro da estrutura Scrum a estimativa não é normalmente feita em termos de tempo – uma métrica mais abstrata de quantificação de esforço é usada. As estimativas mais comuns usadas incluem tamanhos entre 1 e 10, tamanhos de camisetas (PP, P, M, G, GG, GGG) ou a sequência Fibonacci (1,2,3,5,8,13,21,34, etc.).

O importante é que a equipe compartilhe um entendimento comum com a escala que usam, para que cada membro da equipe se sinta confortável ao usá-la.

Planning Poker® / Poker Scrum

Em um ambiente de desenvolvimento de software, constantemente nos deparamos com a necessidade de estimar o esforço necessário para a realização de uma tarefa.

O problema é que numa área de atuação que envolve **trabalho criativo** e soluções baseadas em **esforço subjetivo** como é a área de Desenvolvimento, Design ou Marketing, é bem provável que você terá divergências quanto à estimativa de esforço dada por cada pessoa do seu time.

Para facilitar este processo, **Mike Cohn** em seu livro *Agile Estimating and Planning* [k] fala sobre esta técnica de estimativas de tamanho de tarefas voltadas para as metodologias ágeis e Desenvolvimento de Software.

O Planning Poker® (também chamado de Poker Scrum) se baseia na ideia de estimar ideias através de um jogo de cartas, onde todos os envolvidos no processo devem fazer uma estimativa pessoal com base no seu conhecimento pessoal e na análise de uma tarefa, e o objetivo final é que haja um consenso através de discussão e argumentação rápida para tal.

Ao usar o Planning Poker®, influências entre participantes são minimizadas resultando em uma estimativa mais precisa. [j]

Para se jogar Planning Poker® os seguintes itens são necessários:

- A lista de funcionalidades a serem estimadas;
- Um baralho de cartas numeradas.

Um típico baralho contém a sequência Fibonacci incluindo o zero: 0, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89; outras progressões similares também são possíveis. A razão para usar a sequência Fibonacci é refletir a incerteza em estimar itens grandes. Uma estimativa grande geralmente significa que a história não está bem compreendida em detalhes ou deveria ser dividida em histórias menores. Histórias menores podem ser estimadas em maiores detalhes. Seria um desperdício de tempo discutir se é 19, 20, ou 25; A história é simplesmente (muito) grande.

O jogo pode ser jogado seguindo os seguintes passos:

- O Product Owner apresenta a história a ser estimada. A equipe Scrum faz perguntas e o Product Owner explica em mais detalhes. Se muitas histórias precisam ser estimadas, uma caixa de tempo (ex. somente um minuto por explicação) pode ser

especificada. Se a restrição de tempo for alcançada e a Equipe Scrum ainda não entendeu a estória então é sinal de que a estória precisa ser reescrita;

- Cada membro da Equipe Scrum escolhe secretamente a carta representando sua estimativa;
- Depois que todos escolherem uma carta, todas as seleções são reveladas;
- Indivíduos com estimativas altas ou baixas explicam suas estimativas;
- O jogo se repete até que todas as estórias estejam estimadas;
- Se a discrepância numérica das estimativas for muito grande, possivelmente a estória não tenha sido completamente compreendida;
- O ideal é que todos escolham estimativas iguais ou similares, sendo priorizado o valor numérico da estimativa com maior repetição escolhida (ex. 5 membros da equipe escolheram 3, e 2 membros da equipe escolheram 1, a estimativa usada será a maioria, ou seja, 3).

Gráfico Burndown

O gráfico de Burndown Scrum é uma ferramenta de medição visual que mostra o trabalho completado diariamente em comparação ao ritmo projetado para o lançamento atual do projeto. Seu propósito é permitir que o projeto permaneça dentro do planejado e que o lançamento fique dentro da agenda pretendida.

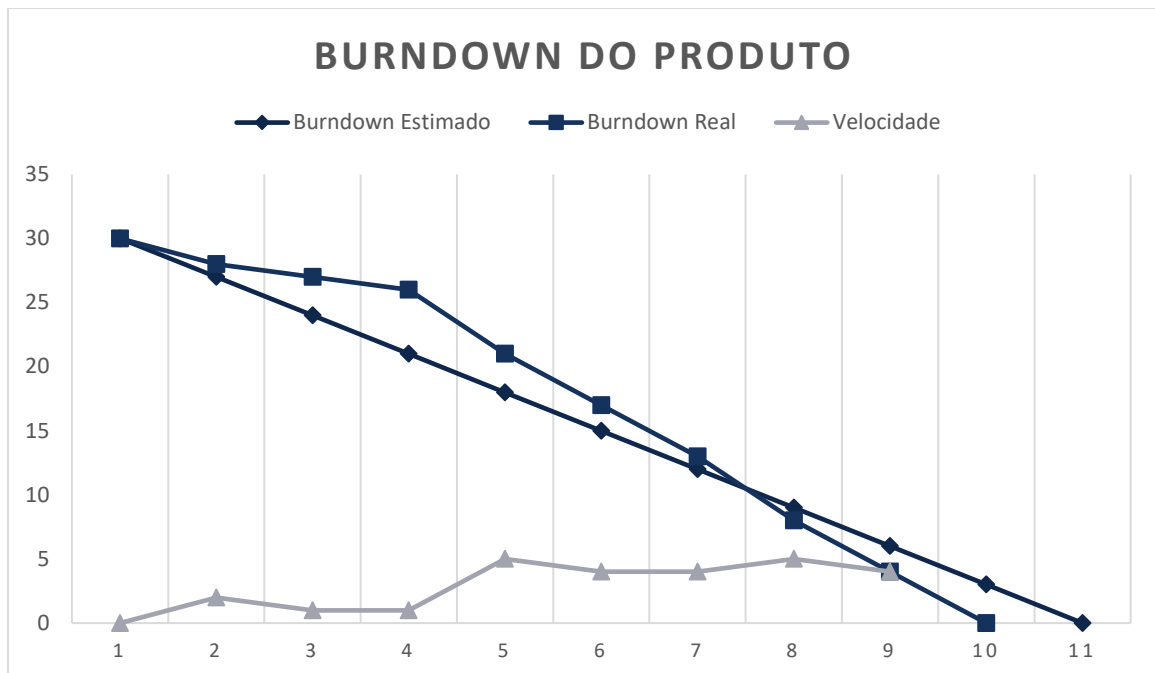


Gráfico simples de Burndown

O ritmo do progresso de uma equipe Scrum é chamada de "velocidade". Essa, expressa a quantidade de estórias ou pontos que foram completados por iteração. Uma regra importante para calcular a velocidade é que somente são contadas as estórias que forem completadas ao final da iteração. A contagem de trabalho parcialmente entregue (ex. somente código sem testes) é estritamente proibida.

Depois de algumas Sprints a velocidade de uma Equipe Scrum certamente será previsível e permitirá uma estimativa mais precisa sobre o tempo necessário até que todos os itens do Backlog do Produto sejam completados. Se a velocidade da Equipe Scrum é de 30 estórias e o total de trabalho restante é 155, poderemos prever que serão necessárias 6 Sprints para completar todas as estórias do Backlog.

Apesar de tudo isso, os itens do Backlog do Produto irão mudar durante a duração do projeto. Novas estórias serão adicionadas e outras estórias serão mudadas ou até

removidas. No gráfico de Burndown simples a velocidade da Equipe Scrum e as mudanças de escopo não podem ser distinguidas. Para refletir isso, outros tipos de diagramas são usados.

Definição de Pronto

Para que se decida quando uma atividade do Sprint Backlog foi completada, a “Definição de Pronto” (DP) é usada. É uma lista de verificação completa com as atividades necessárias para garantir que somente as funcionalidades realmente completadas são entregues, não somente em termos de funcionalidade, mas também de qualidade. A Definição de Pronto pode variar de uma Equipe Scrum para outra, mas deve ser consistente dentro da Equipe. Podem haver “Definições de Pronto” diferentes em vários níveis:

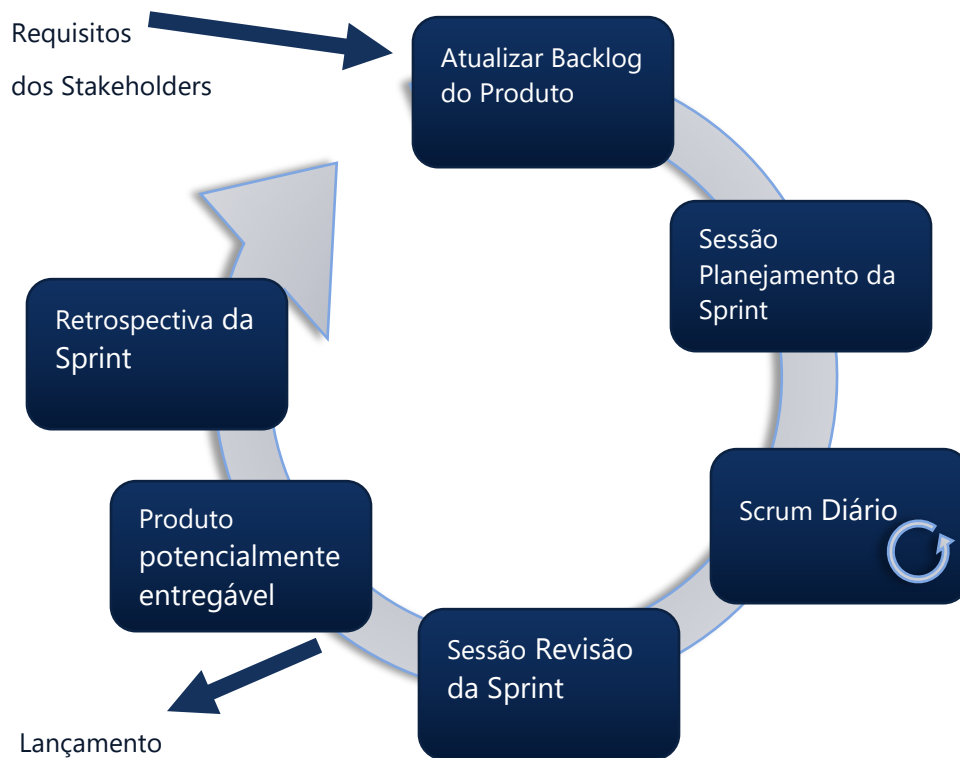
- DP para um item do Backlog do Produto (ex. escrevendo código, testes e toda documentação necessária);
- DP para uma Sprint (ex. instalação do novo sistema demo para revisão);
- DP para lançamentos (ex. redigir notas de lançamento).

Planejamento

Sprint

Na estrutura Scrum todas as atividades necessárias para a implementação dos itens do Backlog do Produto são desempenhadas dentro de uma Sprint (também chamada iteração). Sprints são sempre curtas: normalmente entre 2 a 4 semanas.

Cada Sprint segue um processo definido, como abaixo:



O Processo Sprint

Cada Sprint começa com duas sessões de planejamento para definir o conteúdo da Sprint: a reunião O QUE, e a reunião COMO. A combinação dessas duas reuniões também é conhecida como Reunião de Planejamento da Sprint. Na reunião O QUE a Equipe Scrum se compromete com as histórias do Backlog do Produto e usa a reunião de COMO para dividir essas histórias em tarefas menores e concretas. Então as implementações começam. Ao final da Sprint, uma reunião de Revisão da Sprint é conduzida afim de que o Product Owner possa verificar se todos os itens foram completados e implementados corretamente. Além disso, a Retrospectiva da Sprint é conduzida para verificar e melhorar os processos de execução do projeto: O que foi bom durante a Sprint, o que deve continuar como está e o que deve ser melhorado.

Durante a Sprint uma reunião curta chamada Scrum Diário (ou Reunião de Pé) é feita para atualizar os estados de cada item e para ajudar a auto-organização da equipe.

Reunião de Planejamento de Sprint

Cada Sprint e cada Reunião de Planejamento da Sprint começam com a reunião do O QUE. A meta para essas sessões é definir um Backlog da Sprint realístico que contenha todos os itens que possam ser completamente implementados até o final da Sprint. Essa reunião dura 8 horas para cada 4 semanas de Sprint.

Preparação

Para que uma reunião de O QUE obtenha sucesso, certos preparativos são necessários:

- O Product Owner define a Meta da Sprint;

- Com base na meta os itens relevantes do Backlog do Produto são escolhidos pelo Product Owner;
- Os itens são atualizados e divididos em estórias menores para que sejam completadas durante uma Sprint;
- Os itens são estimados e priorizados;
- A equipe define sua capacidade para a Sprint.

Meta da Sprint

O Product Owner define a meta da Sprint. Essa meta é uma curta descrição do que a Sprint irá tentar alcançar e, deve ser realista e compreensível a todos.

Capacidade da Equipe

A capacidade total da Equipe Scrum pode mudar de uma Sprint para outra. Para que os compromettimentos sejam realistas é necessário saber a capacidade total da equipe para a próxima Sprint considerando férias, feriados, esforços das reuniões de Sprint e tempo necessário para completar outras atividades durante a Sprint.

A sessão de Reunião

Durante a sessão o Product Owner apresenta a Meta da Sprint e discute sobre a mesma com a equipe. Depois disso, a Equipe Scrum itera por todos os itens do Backlog do Produto que sejam relevantes e se compromete com aqueles que pensa serem capazes de completar durante a Sprint. A decisão deve ser baseada na capacidade e compreensão disponíveis sobre os itens.

Ao final da sessão a lista de todos os itens comprometidos do Backlog do Produto dão base para a reunião de COMO e do Backlog da Sprint.

A Reunião de COMO

A meta da reunião de COMO é preencher o Backlog da Sprint identificando as tarefas necessárias a serem completadas para a implementação dos itens do Backlog do Produto. Tarefas normalmente incluem atividades de design, implementação, testes e documentação.

A reunião de COMO pode ser feita separadamente e depois da reunião de O QUE, ou mesmo durante a reunião de O QUE, após se comprometerem aos itens. Ou ambos.

Depois de identificar as atividades necessárias estas são estimadas pela equipe. A base dessa estimativa deve ser horas por pessoa. A equipe deve decidir quanto irá demorar para completar tudo que for requerido a fim de completar cada atividade.

Scrum Diário

A reunião de Scrum diário é curta e acontece todos os dias, de preferência no início do dia de trabalho. Cada membro da equipe que trabalha para completar uma tarefa da Sprint precisa participar. Durante a reunião, cada membro da equipe deve brevemente descrever respostas às seguintes três questões:

- O que ele/ela completou desde a última reunião de Scrum Diário?
- O que ele/ela irá completar até a próxima reunião de Scrum diário?
- Quais são os impedimentos para que ele/ela complete suas tarefas?

Todos os membros da equipe devem participar e devem estar de pé durante a reunião. O ideal é que a reunião de Scrum Diário dure não mais que 15 minutos. Nenhuma dúvida ou problema mencionado durante a reunião são ignorados por falta de tempo. Esses problemas ou dúvidas devem ser registrados pelo Scrum Master e devem ser abordadas mais especificamente após a reunião.

Reunião de Revisão da Sprint

Ao final de cada Sprint ocorre uma Reunião de Revisão da Sprint. Durante essa reunião a Equipe Scrum mostra quais itens do Backlog do Produto foram completados (de acordo com a definição de pronto) durante a Sprint. Isso deve ocorrer em forma de demonstração das novas funcionalidades. Essa reunião geralmente dura 4 horas para cada 4 semanas de Sprint.

É importante notar que os itens do Backlog que não foram completados não devem ser demonstrados. De outra forma isso pode sugerir que os itens estejam completados. Ao invés disso os itens incompletos/remanescentes devem voltar ao Backlog do Produto, estimados novamente e completados em uma Sprint posterior.

A Reunião de Revisão da Sprint deve ser informal. Não se aconselha uso de slides de PowerPoint® tampouco uso de muito tempo para preparação dessa apresentação. Durante a reunião o Product Owner inspeciona os itens de Backlog implementados e aceita a solução, ou adiciona novas estórias ao Backlog do Produto, para adaptar a funcionalidade.

Participantes na reunião de revisão da Sprint geralmente incluem o Product Owner, a Equipe Scrum e o Scrum Master. Ocasionalmente a gerência, clientes ou desenvolvedores de outros projetos podem participar.

Reunião de Retrospectiva da Sprint

Depois que a reunião de Revisão da Sprint terminar, o Scrum Master e a Equipe Scrum se juntam para a Retrospectiva da Sprint. Nessa reunião todos os membros refletem sobre a última Sprint e avaliam três coisas: o que deu certo durante a Sprint; o que não deu certo; e o quais melhorias podem ser feitas na próxima Sprint. Essa reunião deve ter tempo limitado, geralmente. 3 horas para cada 4 semanas de Sprint.

A retrospectiva da Sprint é parte integral do processo de “inspecionar e adaptar”. Sem essa reunião a equipe nunca será capaz de melhorar seu desempenho e não conseguirá focar na melhoria coletiva da equipe. Portanto, sugestões de ações de melhoria de desempenho devem ser disponibilizadas ao final da reunião.

Planejamento de Lançamento (release)

O plano de lançamento é uma definição genérica e de alto nível com base em múltiplas Sprints. Esse plano reflete a expectativa sobre cada funcionalidade que será implementada e quando estarão disponíveis. Também serve como parâmetro de monitoramento do progresso do projeto. Os lançamentos podem ser pequenos pacotes intermediários durante o projeto, ou um pacote completo ao final dessa coleção de Sprints.

Para criar um plano de lançamento é necessário:

- Uma lista priorizada do Backlog do Produto;
- A velocidade (estimada) da Equipe Scrum;
- Condições de satisfação (metas para a agenda, escopo e recursos).

Dependendo do tipo de projeto (funcionalidade ou dados) o plano de lançamento pode ser criado da seguinte forma:

Para um projeto de funcionalidades, por exemplo, a soma de todas as funcionalidades dentro do lançamento pode ser dividida pela velocidade esperada. Isso irá resultar no número de Sprints necessárias para completar as funcionalidades requeridas.

Scrum saudável

Scrum, em seu conceito, significa: Um framework dentro do qual pessoas podem tratar e resolver problemas complexos e adaptativos, enquanto produtiva e criativamente entregam produtos com o mais alto valor possível. Scrum é:

- Leve;
- Simples de entender;
- Extremamente difícil de dominar.

Scrum é um framework estrutural que está sendo usado para gerenciar o desenvolvimento de produtos complexos desde o início de 1990. Scrum não é um processo ou uma técnica para construir produtos; em vez disso, é um framework dentro do qual você pode empregar vários processos ou técnicas. O Scrum deixa claro a eficácia relativa das práticas de gerenciamento e desenvolvimento de produtos, de modo que você possa melhorá-las. [f]

Faz parte da expectativa do Scrum que todo o processo esteja sempre sob constante inspeção e adaptação, a fim de maximizar o desempenho da equipe ao ponto de 100% de sucesso e previsibilidade. Portanto, a norma é estarem sempre fora da zona de conforto:

Zona de conforto é uma série de ações, pensamentos e comportamentos que uma pessoa está acostumada a ter e que não a causam nenhum tipo de medo, ansiedade ou risco. É uma região onde nenhum indivíduo se sente ameaçado.

Na zona de conforto, as pessoas realizam sempre um determinado tipo de comportamento que lhe dá um desempenho constante, porém limitado e com uma sensação de segurança. Essa segurança é uma falsa segurança, uma vez que, quando ocorre uma grande mudança, quem está muito confortável leva um choque maior, e estará menos preparado para sobreviver do que os outros.

Os indivíduos em geral, necessitam saber operar fora de sua zona de conforto para realizar avanços, melhorar seu desempenho seja ele no trabalho, na vida pessoal etc. A zona de conforto é um tema sempre muito debatido na psicologia.

[i]

A melhoria constante impulsiona todos os envolvidos – equipe, Scrum Master e Product Owner – a estarem sempre analisando os eventos, artefatos e processos a fim de encontrarem pontos que funcionam ou não, removendo impedimentos e promovendo colaboração, comunicação e eficácia. Scrum não almeja ditar cada minuto de um projeto, mas sim indicar o caminho ao sucesso.

Revisão

O Que é Scrum?

Scrum é uma estrutura leve que define as atividades, reuniões, funções e artefatos relacionados à projetos geralmente pequenos e de maior incerteza.

Funções Scrum

Product Owner

Indivíduo responsável pelo produto final, gerencia e atualiza o Product Backlog, tornando-o claro e completo para que a equipe Scrum possa implementar as funcionalidades contidas no Backlog do Produto. Junto com o Scrum Master a Equipe Scrum determina a “Definição de Pronto”.

Scrum Master

Mestre-servo da equipe que faz interface entre a equipe e o Product Owner (e outros Stakeholders). Também é responsável por treinar a equipe, remover impedimentos, inspecionar e adaptar a implementação do Scrum.

Equipe Scrum

Equipe multifuncional que não faz distinção de funções, todos são desenvolvedores. Tal equipe é responsável pelas estimativas de esforço dos itens no Backlog do Produto, e por se comprometerem aos itens do Sprint Backlog.

Artefatos Scrum

Backlog do Produto

Repositório de todas as funcionalidades do produto, de acordo com sua prioridade e valor agregado. O Backlog é dinâmico e muda conforme as necessidades do projeto mudam. É atualizado frequentemente pelo Product Owner.

User Stories (Estórias de Usuário)

São explicações curtas e claras sobre a finalidade de cada funcionalidade. Descrita como uma narrativa do usuário final, com uma ação e meta. Cada estória guia a Equipe na estimativa e para definição das atividades que implementam tal estória.

Estimativas de Esforço

As estimativas determinam a dificuldade ou quantidade de esforço exigidas para cada estória. Tal estimativa pode usar variados tipos de unidade de medição, no entanto, nesse nível, não estimam horas de trabalho, somente esforço.

Gráfico Burndown

O Gráfico Burndown auxilia o acompanhamento do progresso do projeto e da velocidade em que a Equipe completa cada estória. A leitura correta desse gráfico auxilia no planejamento de Sprints futuras e nos lançamentos (parciais ou final).

Sprint Backlog

A coleção de tarefas ou atividades necessárias para completar cada estória do Backlog do Produto. O Sprint Backlog contém somente as atividades relacionadas ao Sprint atual e pode ser ajustado conforme as estórias do Backlog do Produto forem atualizadas.

Definição de Pronto

Convenção entre a Equipe Scrum, Scrum Master e Product Owner que determina as validações necessárias para que cada estória esteja completa, possivelmente inclui design, implementação, testes, documentação e lançamento para cada item do Backlog do Produto.

Eventos Scrum

Sprint

Coleção de todas as atividades necessárias para a implementação dos itens do Backlog do Produto. Sprints são sempre curtas: normalmente entre 2 a 4 semanas. Uma Sprint depende de vários eventos que são limitados por tempo.

Reunião de Planejamento de Sprint

Cada Sprint e cada Reunião de Planejamento da Sprint começam com a reunião que determina o que será implementado na próxima Sprint. A meta para essas sessões é definir um Backlog da Sprint realístico que contenha todos os itens que possam ser implementados até o final da Sprint.

Scrum Diário

Uma reunião curta que acontece todos os dias. Todos os membros da equipe devem participar. Durante a reunião, cada membro da equipe deve brevemente descrever respostas às seguintes três questões:

- O que ele/ela completou desde a última reunião de Scrum Diário?
- O que ele/ela irá completar até a próxima reunião de Scrum diário?
- Quais são os impedimentos para que ele/ela complete suas tarefas?

Reunião de Revisão da Sprint

Ao final de cada Sprint ocorre uma Reunião de Revisão da Sprint. Durante essa reunião a Equipe Scrum mostra quais itens do Backlog do Produto foram completados (de acordo com a definição de pronto) durante a Sprint. Isso deve ocorrer em forma de demonstração das novas funcionalidades. As demonstrações devem ser simples e não devem tomar muito tempo. O Product Owner participa dessa reunião para validar que as funcionalidades foram completadas.

Reunião de Retrospectiva da Sprint

Depois da reunião de Revisão da Sprint terminar, o Scrum Master e a Equipe Scrum se juntam para a Retrospectiva da Sprint. Nessa reunião todos os membros refletem sobre a última Sprint e avaliam três coisas: o que deu certo durante a Sprint; o que não deu certo; e quais melhorias podem ser feitas na próxima Sprint. Esse evento é vital para o processo de “inspeção e adaptação” que almeja melhorar a implementação do Scrum no projeto e aumentar o desempenho da equipe.

Atividades

Atividade 1 [30 minutos]

No início do primeiro dia do curso, a dinâmica da **corda** foi aplicada ao grupo, dando a oportunidade de todos testarem sua capacidade de comunicação, liderança e paciência. Escreva (esta atividade **não** será revisada pelo instrutor ou grupo) suas impressões sobre o grupo e sobre você mesmo. Descreva sentimentos, opiniões e conclusões, além de escrever metas que gostaria de melhorar no próximo ano.

Ao final do curso, escreva novamente suas impressões sobre o grupo e você mesmo, agora considerando as lições aprendidas no curso e a convivência com todos durante os 2 dias. Suas metas mudaram?

Atividade 2 [60 minutos]

Crie 10 especificações para um produto. O produto em questão é um portal online de agendamento de audiências judiciais em um fórum civil qualquer. Tais especificações devem levar em consideração dois tipos de usuários finais:

- O cartorário que irá formalizar a audiência no fórum;
- O advogado que irá fazer a petição de audiência via formulário online.

As especificações criadas devem seguir o modelo discutido em Especificação de Requisitos (formato Scrum) [página 12] devem incluir:

- Ator
- Ação
- Conquista

Ator: o 'dono' da estória. Frequentemente um usuário, mas é recomendado que seja mais específico. Ao usar atores específicos (ex. Administrador, Cliente Logado, Visitante não Autenticado) é mais fácil entender e definir o contexto da estória do usuário.

Ação: O que o ator quer fazer. É uma ação mandatória que seja prefixada com "precisa" ou "quer".

Conquista: O que o ator pretende conquistar ou completar ao desempenhar a ação. Esse é o resultado esperado da ação, vista da perspectiva do ator.

Atividade 3 [60-120 minutos]

Usando as especificações criadas na Atividade 1, faça a estimativa de cada uma das 10 estórias usando o Planning Poker®. Para maiores informações sobre o processo do Planning Poker® leia o texto abaixo.

Estória	Estimativa de Esforço

Planning Poker ♠ ♥ ♣ ♦

O Baralho

O baralho de Planning Poker é constituído de uma sequência similar à de Fibonacci, tendo cartas com os seguintes valores 0, 1/2, 1, 2, 3, 5, 8, 13, 20, 40 e 100 havendo ainda uma carta com o símbolo de interrogação, que representa que o estimador não se sente apto a estimar ou participar da discussão, além de uma variação em que aparece uma carta de xícara de café que significa uma sugestão para uma pausa.

O Processo

Durante o processo de Planning Poker, a metodologia permite ter um analista (podendo ser o próprio Scrum Master), que deverá atuar como facilitador da reunião e juiz do processo.

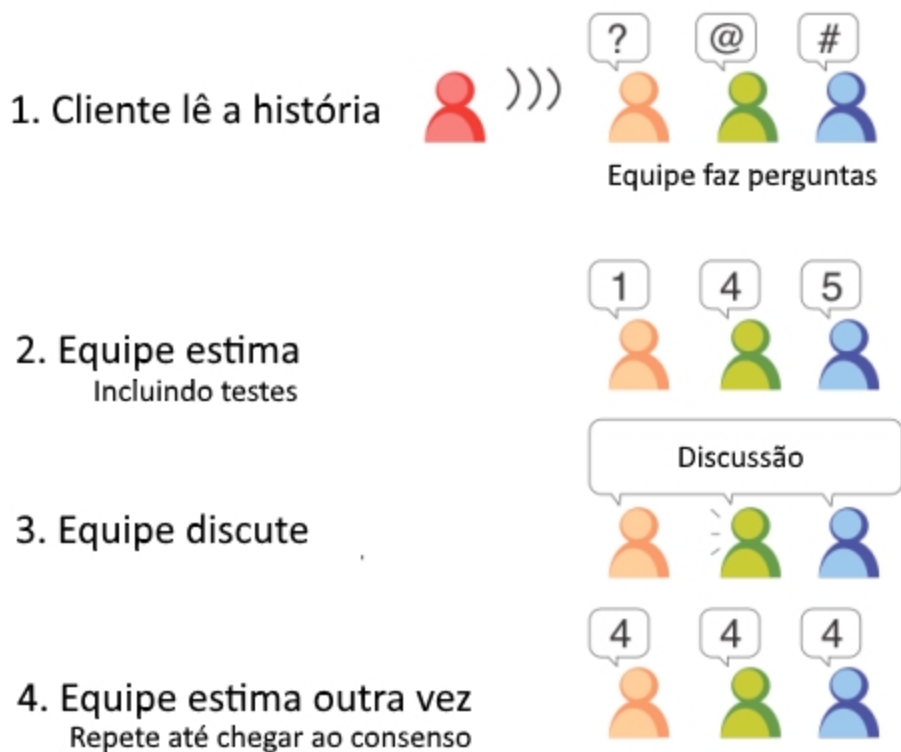
1. O analista deverá reunir uma lista de tarefas para serem estimadas, para isso, ele deve realizar a sua análise, da melhor forma que puder e montar a sequência em que serão discutidas;
2. O analista deve explicar para os membros da equipe a tarefa a ser realizada, levantando as complicações técnicas que ele mesmo tenha identificado, áreas e itens que devem ser levados em conta no processo de desenvolvimento;
3. Após explicar a tarefa, o analista deve fazer uma validação com os membros para saber se todos os envolvidos estão cientes e entenderam o que envolve e o objetivo da tarefa;
4. Inicia-se a discussão de estimativa: cada envolvido deve selecionar o valor de esforço que pensa ser o valor de esforço desta tarefa.
 - a. É importante que o time defina um valor base para sua referência.

Por exemplo

Cadastro simples = 3 pontos

Listagem sem ações = 1 ponto

- b. Nem toda equipe tem os mesmos valores para suas estimativas, e é importante que a equipe refine com frequência estes valores, discutam um consenso.
5. Todos mostram os valores de sua estimativa;
6. Caso todos concordem com o mesmo valor, a estimativa está concluída.



Quando o time **discorda** em suas estimativas, deve-se eleger aquele de estimativa mais alta e o de mais baixa, e estes devem explicar o porquê votaram naqueles valores.

Isto ajuda para que o conhecimento de soluções mais práticas e problemas mais ocultos sejam levados à tona, por exemplo quando um desenvolvedor pensa que uma tarefa pode ser simples e vota 3, mas um testador vota 20 porque consegue pensar em inúmeros casos de teste que o desenvolvedor não chegou a pensar, e isto pode elevar o esforço da tarefa e o tempo de desenvolvimento, mas torna mais assertiva a estimativa gerada.

Outra coisa que é importante ter em mente, é que quando o time **concorda** que uma tarefa possui uma estimativa muito alta, é importante que possa dividir esta grande tarefa em menores, mesmo porquê, o foco do desenvolvimento ágil é a ter entregas constantes e regulares, que agreguem valor ao seu cliente. Outro motivo importante de se dividir uma tarefa em várias menores, é que você poderá criar pequenos checkpoints para seu time, assim, ele tem uma noção melhor do desenvolvimento da demanda, e tem uma sensação

mais aproximada de sua produtividade, influenciando em sua motivação e satisfação no resultado de seu trabalho. [j]

Laboratório de Scrum

Aplicando Estrutura Scrum

Você foi contratado por uma empresa de consultoria médica que foi contratada por uma administradora de hospitais para criar um sistema de gerenciamento de filas e registros médicos, com base no gerenciamento de prioridades Manchester de cores.

Esse sistema deve ser capaz de gerenciar filas de atendimento médico, em emergências e clínicas vinculadas ao sistema hospitalar.

O módulo de registros médicos deve ser capaz de gerenciar todos os registros médicos, exames, diagnósticos, cobranças, histórico médico e dados pessoais dos pacientes, além de bancos de dados sobre medicamentos, exames, artigos médicos e comunicação entre profissionais médicos.

O projeto é novo, com várias incertezas e requisitos ainda em discussão. O cliente final inclui os médicos e profissionais da área que usarão o sistema diariamente, então o sistema tem que ser fácil de aprender e usar, pois seus usuários são muito exigentes.

Planeja-se implementar o projeto em até 12 meses, sendo que os Stakeholders querem ser capazes de acompanhar o desenvolvimento e fazerem mudanças conforme as necessidades. O orçamento planejado é grande, devido à grande visibilidade do projeto, que terá grande impacto na qualidade que o sistema hospitalar será capaz de prover aos seus clientes, aumentando sua visibilidade e valor no mercado de saúde.

Sua organização decide usar a metodologia Scrum para este projeto.

SCRUM FOUNDATION [120 minutos]

Organização

Designe funções específicas para cada membro da Equipe, sendo a maior parte membros da equipe de desenvolvimento, um Product Owner e um Scrum Master.

Cada função deverá cumprir as seguintes designações:

- Product Owner

Criar uma lista de 10 work items (PBI) que serão inseridas no Backlog do produto, com base nos requisitos básicos definidos na descrição da atividade. Esses itens devem conter uma história de usuário simples, mas cada uma com valor do negócio e critério de aceite. Deve estar pronto para explicar os requisitos de cada item para a equipe. Também deve ser capaz de definir a meta de cada Sprint.

- Scrum Master e Equipe de Desenvolvimento

Criem a Definição de Pronto, com base na descrição da atividade. Considerem que cada função da equipe seja igual à dos outros membros, ou seja, não haverá especialistas na equipe – todos sabem como completar todas as atividades.

Criem as fases ou estados de cada atividade e item de trabalho a fim de que sejam gerenciados pela equipe em um task board.

Criem um task board usando o conceito Kanban e definam a capacidade de cada membro da equipe.

- Scrum Master

O Scrum Master deve auxiliar a equipe a escolher os requisitos da definição de pronto que sejam realistas e alcançáveis, e que mantenha a equipe em foco facilitando e mediando os eventos Scrum.

Deve gerenciar o gráfico de Burndown da Sprint, atualizando a velocidade diária da equipe e as atividades completadas.

Artefatos

Em cada dia da Sprint, o Scrum Master deve atualizar o Gráfico de Burndown.

Após cada reunião de Scrum Diário, cada membro da Equipe deve atualizar o Task Board com o estado das atividades.

Cada atividade que for completada deve ter finalizado 100% da definição de pronto.

Eventos

Toda equipe + Scrum Master + Product Owner devem participar da reunião de planejamento da Sprint. A primeira fase dessa reunião conta com as explicações do PO para compreender os requisitos e propósitos de cada item de trabalho (PBI). O PO compartilha a meta da Sprint. Nesta Etapa, o PO, Scrum Master e Equipe Scrum definem o tempo total da Sprint (2-4 semanas)

Após compreenderem cada item de trabalho, a equipe se reúne para segunda fase da reunião de planejamento da Sprint, em que usam o planning poker para estimar o esforço de cada item de trabalho (PBI). Após definirem o esforço de cada uma, quebram cada PBI em atividades que serão necessárias para completar cada item de trabalho e seja considerado pronto.

Após a reunião de planejamento da Sprint. A Equipe inicia a primeira Sprint com a primeira reunião de Scrum Diário, definindo quais tarefas serão completadas naquele dia, e por quem.

Durante aquele "dia", cada membro da equipe "trabalha" em sua atividade, ou aponta os impedimentos para o Scrum Master. Entre esses impedimentos, aponte ferramentas, frameworks ou dificuldades que precisam ser tratados durante esse dia.

Durante cada Scrum Diário, a Equipe atualiza o Task Board com o progresso de cada atividade, até serem completadas.

Ao final da Sprint, a Equipe convida o PO novamente para "apresentar" os PBI que estiverem prontos, apresentando as funcionalidades que foram implementadas.

Após a reunião de Revisão da Sprint, a Equipe e o Scrum Master se reúnem para a Retrospectiva da Sprint, onde serão discutidos todos os impedimentos, o que foi feito para resolvê-los, o que deu certo e o que pode ser melhorado na próxima Sprint. O Scrum Master ativamente incentiva cada membro a compartilhar sua opinião.

Avaliação

Questão 1

Qual é o propósito da Retrospectiva da Sprint? Selecionar as opções que NÃO aderem ao propósito desta reunião importante:

- a) Discutir os impedimentos encontrados pela Equipe de Desenvolvimento durante a última Sprint, e fazer um plano de melhorias.
- b) Refinar os épicos designados pelo Product Owner para as próximas duas Sprints, para promover um planejamento de lançamento mais confiável.

- c) Verificar como que o incremento do produto satisfaz as estórias definidas no Backlog do Produto.
- d) Discutir a interação dentro da Equipe Scrum, e acordar medidas de melhoria na comunicação.

Questão 2

A função de Product Owner e Scrum Master não estão inclusas na contagem da Equipe de Desenvolvimento.

- a) Verdadeiro.
- b) Falso.

Questão 3

Scrum permite que tarefas sejam estimadas novamente com base num crescimento futuro. Quais membros da equipe Scrum é responsável por atualizar a estimativa do trabalho durante uma Sprint?

- a) A Equipe de Desenvolvimento.
- b) O Scrum Master.
- c) O membro mais sênior da Equipe.
- d) O Product Owner.

Questão 4

Time-boxing é um princípio importante de Scrum. Qual é o significado de uma reunião com timebox?

- a) A reunião deve acontecer até um certo horário.
- b) A reunião deve acontecer no mesmo horário todos os dias.
- c) A reunião deve estender-se por pelo menos um tempo mínimo.
- d) A reunião não pode estender-se por um tempo máximo.

Questão 5

Resolver conflitos internos NÃO é responsabilidade da Equipe de Desenvolvimento

- a) Verdadeiro.
- b) Falso.

Questão 6

O que NÃO é um atributo da Equipe de Desenvolvimento?

- a) Membros das Equipes de Desenvolvimento são transferidos/trocados com frequência para promover o aprendizado contínuo e multifuncionalidades.
- b) A Equipe de Desenvolvimento provê comentários durante a Reunião de Planejamento de Sprint com respeito à capacidade do projeto durante a próxima Sprint.
- c) A Equipe de Desenvolvimento pode renegociar, com o Product Owner, o trabalho a ser entregue e firmam acordo sobre a meta da Sprint durante uma Sprint corrente/atual, quando mais informações são recebidas.
- d) A Equipe de Desenvolvimento atualiza suas estimativas do total de trabalho restante para terminarem a Sprint atual, afim de que possa ser inserida no Gráfico de Burndown da Sprint.

Questão 7

Um dos benefícios de Scrum é que a Equipe de Desenvolvimento não tem que escrever mais especificações detalhadas.

- a) Verdadeiro.
- b) Falso.

Questão 8

Quais são as maiores propriedades de uma Equipe de Desenvolvimento multifuncional?

- a) A Equipe é capaz de completar o projeto de acordo com o plano, depois que se comprometem com a data e custos junto ao Product Owner.
- b) A Equipe tem todas as habilidades disponíveis e necessárias para aceitar responsabilidade coletiva pelo próximo incremento do produto.
- c) A Equipe conta com membros competentes e dedicados a um domínio em particular, como testes especializados ou análise de negócios, para facilitar a entrega do maior valor possível ao negócio.

Referencias

- [a]** <https://zenexmachina.wordpress.com/2012/07/19/waterfall-vs-agile-a-knowledge-problem-not-a-requirements-problem/>
- [b]** Polanyi, M. (1958) Personal Knowledge
- [c]** Nonaka, I. & Takeuchi, H. (1995). The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation
- [d]** Tuckman, B. W. Developmental sequence in small groups. Psychological bulletin, v. 63, n. 6, p. 384, 1965.
- [e]** <http://agilemanifesto.org/>
- [f]** <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-Portuguese-Brazilian.pdf>
- [g]** https://pt.wikipedia.org/wiki/Modelo_de_Tuckman
- [h]** <http://www.scrumguides.org/history.html>
- [i]** <http://www.significados.com.br/zona-de-conforto/>
- [j]** <http://blog.acelerato.com/agile/utilizando-planning-poker-para-estimar-seu-backlog/>
- [k]** <https://www.amazon.com.br/Agile-Estimating-Planning-Mike-Cohn/dp/0131479415>
- [l]** <https://www.scrum.as/docs/International%20Scrum%20Master%20Foundation%20-%20Study%20guide.pdf>
- [m]** <http://mplaza.pm/downloads/Scrum%20Training%20Manual.pdf>