Strecht Simulados · Guia de preparação



Guia de preparação PSPO I

Tudo o que a prova cobra, organizado por domínio: conceitos-chave, comparações que confundem, pegadinhas reais, gatilhos do enunciado, questões comentadas e um plano para chegar pronto ao exame.

5 capítulos por domínio	**32** questões comentadas	**85%** nota de corte
6 provas completas no curso		

ANTES DE COMEÇAR

Como usar este guia

Este guia foi feito para ser estudado junto com as 6 provas completas do curso PSPO I: o guia explica o que cai e onde estão as armadilhas; as provas treinam o ritmo e mostram o que você ainda não domina.

Quando	O que fazer
Semana 1	Faça o teste diagnóstico (página seguinte) e leia os capítulos 1 e 2. No fim da semana, faça a Prova 1 do curso no tempo do exame.
Semana 2	Leia os demais capítulos, começando pelos domínios em que você errou mais no diagnóstico. Faça as provas 2 e 3.
Semana 3	Releia os alertas de pegadinha e as tabelas de comparação. Faça as provas 4 e 5 e revise cada erro pela explicação.
Semana 4	Estude a folha de cola, faça a Prova 6 e refaça as provas abaixo de 80%. Agende o exame quando estiver acima de 80% em todas (a nota de corte é 85%).

Mapa do exame

Cada prova do curso tem 80 questões e 60 minutos, no formato do exame. Referência de aprovação: 85%.

Domínio (capítulo)	Peso	Neste guia
1. Gestão do Product Backlog	—	17 conceitos · 14 pegadinhas
2. O Framework Scrum	—	17 conceitos · 14 pegadinhas
3. Responsabilidades do Product Owner	—	17 conceitos · 13 pegadinhas
4. Stakeholders, Previsão e Entregas	—	18 conceitos · 13 pegadinhas
5. Valor, Visão e Estratégia de Produto	—	18 conceitos · 13 pegadinhas

Como ler os capítulos: os **tópicos-chave** são o que mais cai; a **tabela de memorização** separa o que a prova costuma confundir; os **alertas de prova** mostram as armadilhas das alternativas erradas; os **gatilhos** ligam palavras do enunciado à resposta; e o **checklist** diz se você está pronto para o domínio.

Estude um pouco todo dia, de graça

Instagram @strecht.simulados

Um post de prova por dia: questão comentada, pegadinhas e o “não confunda” de cada certificação.

Seguir no Instagram →

YouTube Strecht Simulados

Shorts diários com pegadinhas de prova e, aos sábados, um simulado comentado.

Inscrever-se no canal →

DIAGNÓSTICO

Você já sabe isso?

Responda às 12 questões sem consultar nada. As respostas, com a explicação e o capítulo para revisar, estão no fim do guia. Comece o estudo pelos capítulos em que errou.

D1 · Gestão do Product Backlog

Qual é o objetivo principal de ordenar o Product Backlog?

A. Seguir a ordem cronológica de chegada dos pedidos
B. Garantir que cada Developer tenha trabalho na sua especialidade
C. Distribuir o trabalho igualmente entre as áreas da empresa
D. Maximizar o valor entregue e o aprendizado ao longo das Sprints

D2 · Gestão do Product Backlog

O Product Owner mantém dois documentos com prioridades diferentes: um roadmap aprovado e o Product Backlog. Qual é o problema?

A. Nenhum, pois o roadmap define a ordem e o backlog a execução
B. Duas fontes de prioridade confundem o time e reduzem a transparência
C. Os Developers devem escolher qual dos dois documentos seguir
D. O Scrum Master precisa conciliar os dois documentos no início de cada Sprint Planning

D3 · Gestão do Product Backlog

Os Developers não conseguem estimar um item porque há muita incerteza técnica. Qual é uma abordagem adequada?

A. Deixar o item de fora do Product Backlog até que tudo fique claro para o time
B. Pedir ao Product Owner que estime o item no lugar deles
C. Criar um item menor de investigação para reduzir a incerteza
D. Estimar um número qualquer e ajustar depois da Sprint

D4 · O Framework Scrum

Para quem o Scrum Team apresenta os resultados do trabalho na Sprint Review?

A. À diretoria financeira
B. Aos gestores de cada Developer
C. Aos auditores externos
D. Aos principais stakeholders

D5 · O Framework Scrum

Durante a Sprint, o time percebe que um processo está gerando resultados fora dos limites aceitáveis. Segundo o Scrum Guide, quando o ajuste deve ser feito?

A. Na próxima Sprint Retrospective, o evento dedicado a adaptações
B. Ao final do trimestre, após aprovação da gerência
C. Depois da próxima release, para não atrasar a entrega
D. O quanto antes, para minimizar desvios adicionais

D6 · O Framework Scrum

Uma diretora pergunta ao Product Owner em que o Scrum se baseia para lidar com a incerteza no desenvolvimento de um novo aplicativo. Qual resposta está alinhada ao Scrum Guide?

A. Em métricas de produtividade individual, comparadas entre os membros a cada Sprint
B. No empirismo e no pensamento enxuto: decisões baseadas no que é observado e foco no essencial
C. Em um plano detalhado elaborado no início e revisado a cada mudança formal de escopo
D. Na experiência do Product Owner, que define as soluções e distribui as tarefas ao time

D7 · Responsabilidades do Product Owner

Na Sprint Planning, o Product Owner propõe um objetivo, mas os Developers acham que ele não é viável na Sprint. O que acontece?

A. A Sprint Planning é adiada até haver consenso total
B. O Product Owner mantém a meta e os Developers aceitam
C. O Scrum Team ajusta a Meta da Sprint em conjunto
D. O Scrum Master decide se a meta é viável ou não

D8 · Responsabilidades do Product Owner

O Product Owner prefere não participar da Sprint Planning porque confia nos Developers. Qual é o problema?

A. A Sprint Planning perde a proposta de valor e a conversa sobre os itens
B. O Scrum Master terá de assumir a ordem do backlog
C. A Sprint terá de ser mais curta que o previsto
D. Nenhum, pois a Sprint Planning é um evento dos Developers com o Scrum Master

D9 · Responsabilidades do Product Owner

O Product Owner pode alterar sozinho a Definição de Pronto?

A. Sim, desde que comunique a mudança aos stakeholders na Sprint Review seguinte
B. Não; somente o Scrum Master pode alterar a Definição de Pronto
C. Não; ela é criada pelo Scrum Team ou vem do padrão da organização
D. Sim; ela é um artefato sob responsabilidade do Product Owner

D10 · Stakeholders, Previsão e Entregas

O produto está sujeito a exigências regulatórias. Como garantir que cada Incremento atenda a elas?

A. Registrar as exigências apenas no roadmap do produto
B. Incluir os critérios regulatórios na Definição de Pronto
C. Deixar as verificações para uma Sprint de conformidade antes do lançamento
D. Pedir a um auditor que aprove cada item durante a Daily Scrum

D11 · Stakeholders, Previsão e Entregas

Um cliente importante quer ver o produto a cada duas semanas, mas as Sprints são de um mês. Qual alternativa é coerente com o Scrum?

A. Fazer duas Sprint Reviews extras por Sprint só para esse cliente
B. Pedir ao Scrum Master que envie relatórios quinzenais
C. Avaliar Sprints mais curtas ou entregas antes do fim da Sprint
D. Recusar o pedido, pois a duração da Sprint nunca muda

D12 · Stakeholders, Previsão e Entregas

Qual é o risco de o Product Owner prometer datas aos stakeholders sem consultar os Developers?

A. Reduzir a necessidade de refinamento do backlog
B. Criar expectativas irreais e pressão sobre a qualidade
C. Aumentar a transparência sobre o trabalho do time perante os stakeholders
D. Fortalecer o auto-gerenciamento dos Developers

01

CAPÍTULO 1

Gestão do Product Backlog

Aqui eu preciso dominar o Product Backlog como única fonte de trabalho do Scrum Team: lista emergente, ordenada pelo Product Owner e refinada continuamente rumo à Meta do Produto. A prova cobra muito quem faz o quê: o Product Owner decide conteúdo e ordem, os Developers dimensionam, o Scrum Master não ordena, não estima e não escreve itens. Os cenários giram em torno de refinamento (quando, quem, quanto), divisão de itens grandes, itens técnicos e dívida técnica, vários times no mesmo produto e mudanças durante a Sprint. Onde mais erro: cair em práticas complementares tratadas como obrigatórias (histórias de usuário, pontos de história, MoSCoW, Definition of Ready, regra de 10%) e confundir Product Backlog com Sprint Backlog.

Neste capítulo
17 tópicos-chave · 14 pegadinhas · 7 macetes · 13 gatilhos · 4 questões comentadas

Tópicos-chave

Product Backlog

Lista emergente e ordenada do que é necessário para melhorar o produto e única fonte do trabalho realizado pelo Scrum Team. Contém tudo: funcionalidades, itens técnicos, defeitos, pesquisa e experimentos. Não se limita às próximas Sprints e nunca é congelado; pode ser ajustado pelo Product Owner a qualquer momento.

Meta do Produto (Product Goal)

É o compromisso do Product Backlog e fica dentro dele; o restante do backlog emerge para definir o que vai cumpri-la. É o objetivo de longo prazo do Scrum Team, uma por vez: o time cumpre ou abandona uma meta antes de assumir a próxima. Desenvolvê-la e comunicá-la explicitamente é responsabilidade do Product Owner.

Refinamento do Product Backlog

Atividade contínua (não é evento) de dividir e definir melhor os itens em itens menores e mais precisos, ao longo da Sprint, conforme a necessidade. Adiciona descrição, ordem e tamanho. O Scrum Guide 2020 não define reunião, frequência, duração nem percentual de tempo: o Scrum Team decide como refinar.

Dimensionamento (sizing)

Os Developers que farão o trabalho são responsáveis pelo tamanho dos itens. O Product Owner influencia ajudando a entender o item e oferecendo trade-offs, nunca impondo números. Estimativa é previsão, não compromisso, e muda conforme se aprende.

Ordenação do Product Backlog

Decisão do Product Owner para maximizar valor e aprendizado, considerando valor, tamanho, risco, dependências e urgência. O Scrum Guide não prescreve técnica (MoSCoW, WSJF) nem obriga ordenar só por valor de negócio. O backlog precisa ser ordenado, e não apenas classificado em níveis de prioridade.

Item pronto para seleção

Segundo o Scrum Guide 2020, é o item que pode ser concluído (Done) pelo Scrum Team dentro de uma Sprint. Definition of Ready não existe no Scrum Guide; alternativa que a torna obrigatória está errada. Item grande no topo deve ser refinado e dividido, nunca resolvido estendendo a Sprint.

Detalhe decrescente

Itens mais bem ordenados costumam ser mais claros e detalhados; os do fim podem ser grandes e vagos até se aproximarem do topo. Detalhar centenas de itens para meses ou anos é desperdício, pois muitos vão mudar ou perder o sentido. Basta refinar o suficiente para os Developers selecionarem itens prontos para a próxima Sprint, com pequena folga.

Um produto, um Product Backlog, um Product Owner

Vários Scrum Teams no mesmo produto compartilham a mesma Meta do Produto, o mesmo Product Backlog e o mesmo Product Owner. Dependências entre times são tratadas no refinamento do backlog único, com os times e o Product Owner. Backlogs separados, um terceiro backlog ou um Product Owner por time são sempre errados; já três produtos distintos pedem três backlogs e três Product Owners.

Seleção de itens na Sprint Planning

Os Developers selecionam os itens, em conversa com o Product Owner, guiados pela ordem de valor e pela Meta da Sprint. Ficam mais confiantes conhecendo desempenho passado, capacidade e Definição de Pronto. Stakeholders, Scrum Master e o próprio Product Owner sozinho não selecionam.

Sprint Backlog x mudanças na Sprint

O Sprint Backlog é dos Developers e só muda por decisão deles; mudar o Product Backlog não altera o Sprint Backlog automaticamente. Durante a Sprint o escopo pode ser esclarecido e renegociado com o Product Owner, mas a Meta da Sprint fica protegida, a duração não muda e a qualidade não diminui.

Divisão em fatias utilizáveis

Itens grandes devem ser divididos em fatias que atravessam as camadas técnicas e entregam algo utilizável ao usuário, por fluxo do usuário ou regra de negócio (por exemplo, primeiro o pagamento com cartão, depois os outros meios). Dividir por camada (interface, lógica, banco), por fase (análise, código, testes) ou por especialidade não gera Incremento utilizável.

Dívida técnica e itens técnicos

Trabalho técnico (atualizar bibliotecas, automação de testes, desempenho, dívida técnica) entra no Product Backlog de forma visível e é ordenado pelo Product Owner, ouvindo os Developers sobre valor e risco. Nunca fica em backlog técnico paralelo, embutido em estimativas, fora do horário ou em Sprint técnica. Investir em qualidade aumenta a Capacidade de Inovar.

Itens de investigação (pesquisa)

Quando a incerteza técnica impede estimar, cria-se um item menor de investigação para aprender primeiro. Pesquisa e experimentos são itens do Product Backlog, ordenados pelo Product Owner. Sprint Zero, Sprint de requisitos e trabalho fora da Sprint não existem no Scrum.

Requisito transversal e Definição de Pronto

Requisito que vale para todo o produto, como desempenho ou segurança, costuma ir para a Definição de Pronto, que se aplica a todo o Incremento. Requisito de um recurso específico é item do Product Backlog. Um item só vira parte do Incremento quando atende à Definição de Pronto.

Critérios de aceitação

São por item e complementares: ajudam a dar clareza, mas não são exigidos pelo Scrum Guide 2020. A Definição de Pronto, ao contrário, é obrigatória e vale para todo o Incremento. Não existe Definição de Pronto por item criada pelo Product Owner.

Transparência do Product Backlog

O Product Owner garante que o Product Backlog seja transparente, visível e compreendido pelo Scrum Team e pelos stakeholders. As decisões dele ficam visíveis no conteúdo e na ordem do backlog. Remover ou rebaixar um pedido de stakeholder exige explicar os critérios de valor.

Práticas complementares

Histórias de usuário, pontos de história, planning poker, MoSCoW, critérios de aceitação e Definition of Ready são opcionais. O Scrum Guide cita tamanho, mas não define unidade nem técnica, e também não define ferramenta para o backlog. Alternativa que transforma qualquer delas em regra do Scrum está errada.

Macetes para memorizar

Papéis no backlog
Sugerir é aberto; decidir é do Product Owner; estimar é de quem faz.

Escala
Um produto = um Product Backlog = um Product Owner, com quantos Scrum Teams forem.

Atributos do refinamento
D-O-T: Descrição, Ordem e Tamanho.

Nível de detalhe
Topo fino e claro, fundo grosso e vago: detalhe proporcional à proximidade de execução.

Divisão de itens
Corte o bolo em fatias (cada fatia tem todas as camadas), nunca em camadas.

Compromissos dos artefatos
Backlog com Meta do Produto, Sprint Backlog com Meta da Sprint, Incremento com Definição de Pronto: cada artefato tem o seu.

Mudança na Sprint
Meta fixa, qualidade fixa, duração fixa; escopo negociável.

Tabela de memorização: não confunda

Quando é...	...e quando é
Refinamento do Product Backlog Atividade contínua, sem timebox, que prepara e detalha itens ao longo da Sprint (Product Owner e Developers).	**Sprint Planning** Evento com timebox que inicia a Sprint e define por que ela é valiosa, o que será feito e como (todo o Scrum Team).
Product Backlog Escopo do produto, ordenado pelo Product Owner, com compromisso na Meta do Produto; muda a qualquer momento.	**Sprint Backlog** Plano da Sprint, dos Developers e para os Developers, com compromisso na Meta da Sprint; só os Developers alteram.
Critérios de aceitação Clareza de um item específico; prática complementar, não exigida.	**Definição de Pronto** Padrão de qualidade obrigatório para todo o Incremento, incluindo requisitos transversais como desempenho e segurança.
Fatia vertical (fluxo do usuário ou regra de negócio) Quando a pergunta busca valor e feedback mais cedo: cada parte é utilizável.	**Divisão por camada, fase ou especialidade** Sempre alternativa errada: as partes isoladas não são utilizáveis e adiam o valor.

Backlog ordenado O que o Scrum Guide exige: uma ordem única que torna as decisões transparentes.	**Backlog priorizado em categorias** Níveis como alta, média, baixa ou MoSCoW são técnicas opcionais e não substituem a ordem.
Estimativa Previsão dos Developers, revista conforme o entendimento melhora.	**Compromisso** Só três no Scrum: Meta do Produto, Meta da Sprint e Definição de Pronto.
Roadmap Previsão que dá direção e deve ser coerente com o backlog; não é ordem paralela.	**Product Backlog** Única fonte de trabalho e da ordem; duas fontes de prioridade confundem o time e reduzem a transparência.

Números que caem

Item	Valor
Atributos citados no Scrum Guide 2020 que o refinamento adiciona aos itens	**3: descrição, ordem e tamanho (os atributos variam conforme o domínio)**
Percentual de tempo para refinamento no Scrum Guide 2020	**Nenhum (a regra de 10% é de versões antigas)**
Limite de itens no Product Backlog	**Nenhum definido pelo Scrum Guide**
Metas do Produto perseguidas ao mesmo tempo	**1**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Refinamento acontece em uma Sprint dedicada a requisitos, em Sprint Zero ou em reunião fixa no primeiro dia da Sprint.
Como não cair: Refinamento é contínuo, ao longo da Sprint, conforme a necessidade. Toda Sprint gera Incremento utilizável e o Scrum Guide não define reunião fixa de refinamento.

⚠ **Alerta de prova:** O refinamento tem timebox de 10% da capacidade da Sprint.
Como não cair: A regra de 10% era de versões antigas e não aparece no Scrum Guide 2020. Refinamento não é evento e não tem timebox.

⚠ **Alerta de prova:** O Product Owner deveria estimar os itens porque conhece o valor, ou o Scrum Master porque facilita a sessão.
Como não cair: Quem faz, estima: os Developers dimensionam. O Product Owner ordena e explica; o Scrum Master pode facilitar, mas não estima.

⚠ **Alerta de prova:** Na Sprint Planning, o Product Owner seleciona os itens de acordo com a Meta do Produto.
Como não cair: O Product Owner ordena e propõe valor; a seleção é dos Developers, em conversa com ele, porque eles conhecem a própria capacidade.

⚠ **Alerta de prova:** Vários times no mesmo produto: cada time com seu backlog sincronizado pelos Scrum Masters, ou um backlog único com um Product Owner por time.
Como não cair: Um produto tem um Product Backlog e um Product Owner, independentemente do número de Scrum Teams. Autonomia é no como, não em backlogs separados.

⚠ **Alerta de prova:** Alterar o Product Backlog durante a Sprint exige aprovação do Scrum Master e dos Developers, ou ele fica congelado até a Sprint Review.
Como não cair: O Product Owner pode reordenar o Product Backlog a qualquer momento, sem aprovação. Só o Sprint Backlog é protegido, e muda pelos Developers.

⚠ **Alerta de prova:** Item mais complexo que o previsto: reduzir testes, estender a Sprint ou cancelá-la para reestimar.
Como não cair: Renegocia-se o escopo com o Product Owner. Qualidade não diminui, a Sprint não se estende e só é cancelada se a Meta da Sprint ficar obsoleta.

⚠ **Alerta de prova:** Item com muita incerteza: deixá-lo fora do backlog, pedir ao Product Owner que estime ou chutar um número e ajustar depois.
Como não cair: Crie um item menor de investigação para reduzir a incerteza e então estime. Inflar a estimativa também esconde a incerteza e reduz a transparência.

⚠ **Alerta de prova:** Dívida técnica embutida sem aviso nas estimativas, feita fora do horário ou em uma Sprint técnica trimestral.
Como não cair: Dívida técnica é trabalho visível no Product Backlog, discutido e ordenado com o Product Owner. Sprint técnica não existe.

⚠ **Alerta de prova:** Sugestão de stakeholder na Sprint Review entra direto no topo, ou só pode ser incluída na próxima Sprint Planning.
Como não cair: O backlog pode ser ajustado no próprio evento, mas a posição do item é decidida pelo Product Owner com base em valor.

⚠ **Alerta de prova:** A Meta do Produto é o primeiro item do backlog e deve ser concluída em uma Sprint, ou fica fora dele em documento de visão.
Como não cair: A Meta do Produto está no Product Backlog como compromisso e é objetivo de longo prazo; o time trabalha com uma por vez.

⚠ **Alerta de prova:** Meta do Produto perdeu o sentido, mas só pode ser trocada depois de cumprida; ou se cria uma segunda meta em paralelo.
Como não cair: O Product Owner pode abandoná-la e definir nova meta, comunicando ao time e aos stakeholders. Nunca duas metas ao mesmo tempo.

⚠ **Alerta de prova:** Itens futuros precisam estar tão detalhados quanto os do topo, ou ficam fora do backlog em um roadmap.
Como não cair: O Product Backlog vai além das próximas Sprints com detalhe decrescente. Detalhar tudo é desperdício e roadmap não substitui o backlog.

⚠ **Alerta de prova:** A Definição de Pronto é verificada na Sprint Review, ou cada item tem sua própria Definição de Pronto criada pelo Product Owner.
Como não cair: A Definição de Pronto é aplicada durante o trabalho e vale para todo o Incremento. Item que não a atende nem é apresentado na Sprint Review.

Como cai na prova

📝 **Como cai:** Cenário em que alguém (stakeholder, gerente, Scrum Master, comitê) quer decidir a ordem ou editar o backlog.
Como resolver: Identifique o dono: ordem e conteúdo são do Product Owner. Outros sugerem e tentam convencê-lo. Elimine alternativas em que Scrum Master, comitê ou Developers ordenam.

📝 **Como cai:** Pergunta se o Scrum Guide exige uma técnica, unidade ou formato (histórias, pontos, MoSCoW, DoR, 10%).
Como resolver: Regra geral: o Scrum Guide define o que deve existir, não como fazer. Marque a alternativa que diz que é prática complementar ou decisão do time.

📝 **Como cai:** Problema descoberto durante a Sprint: item complexo, troca de item, pedido urgente.
Como resolver: Verifique se a Meta da Sprint continua válida. Se sim, negocie escopo com os Developers e o Product Owner, sem reduzir qualidade e sem estender a Sprint. Pedido novo sem relação com a meta vai para o Product Backlog.

📝 **Como cai:** Escolha DUAS sobre Product Backlog, refinamento ou Meta do Produto.
Como resolver: Procure as palavras do Scrum Guide: emergente, ordenado, única fonte, atividade contínua, itens menores e mais precisos, objetivo de longo prazo, cumprir ou abandonar. Descarte timebox, Scrum Master conduzindo e várias metas em paralelo.

📝 **Como cai:** Como ordenar dois itens com valor, custo, risco ou dependências diferentes.
Como resolver: Valor sozinho não basta: compare valor com tamanho, risco, urgência e dependências. Ordem de chegada, número de pedidos, preferência técnica e especialidade do Developer nunca são critério.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
Quando o refinamento deve acontecer? Quem decide como refinar?	**Atividade contínua durante a Sprint; técnicas, frequência e duração decididas pelo Scrum Team.**
Quem dimensiona / estima os itens?	**Os Developers que farão o trabalho.**
Item no topo há várias Sprints e nunca é selecionado	**Precisa ser refinado ou ter o valor reavaliado.**
Valor parecido, mas um item reduz grande risco técnico	**Antecipar o item de risco: risco e aprendizado desempatam.**
Item de alto valor depende de outro de baixo valor	**O de baixo valor sobe antes, porque habilita o valioso.**
Vários Scrum Teams no mesmo produto, dependências entre itens	**Um Product Backlog, um Product Owner, refinamento conjunto.**
Requisito de desempenho ou segurança que vale para todas as funcionalidades	**Definição de Pronto.**
Defeito relatado em funcionalidade já liberada	**Vai para o Product Backlog e é ordenado pelo Product Owner conforme o impacto.**
Item grande demais no topo / dividir para entregar valor cedo	**Refinar em fatias utilizáveis por fluxo do usuário ou regra de negócio.**
O Scrum Guide exige histórias de usuário, pontos de história, MoSCoW, critérios de aceitação ou ferramenta?	**Não: práticas complementares; o Scrum Guide não define técnica, unidade nem ferramenta.**
Stakeholders editam o backlog e mudam a ordem diretamente	**Responsabilidade diluída: a ordem deixa de refletir as decisões de valor do Product Owner.**
Item descreve uma solução técnica específica em vez do problema	**Risco de limitar soluções melhores dos Developers: escreva o problema e o valor.**
Centenas de itens antigos ou item que ninguém sabe por que existe	**Reavaliar e remover o que não contribui para a Meta do Produto.**

🎯 Se você lembrar de só uma coisa deste capítulo...

- O Product Backlog é a única fonte de trabalho: itens técnicos, defeitos e pesquisa entram nele, nunca em listas paralelas.
- Os Developers dimensionam e selecionam; o Product Owner ordena e decide o conteúdo; o Scrum Master não ordena nem estima.
- Refinamento é atividade contínua, não evento, sem timebox e sem regra de 10% no Scrum Guide 2020.
- Item pronto para seleção é o que pode ficar Done dentro de uma Sprint; Definition of Ready não está no Scrum Guide.
- Um produto tem um Product Backlog e um Product Owner, mesmo com vários Scrum Teams.

Essenciais para a prova (checklist)

☐ Sei explicar por que o Product Backlog é emergente, ordenado e a única fonte de trabalho do Scrum Team.

☐ Sei que a Meta do Produto está dentro do Product Backlog, é uma por vez e pode ser abandonada.

☐ Sei quem dimensiona, quem ordena e quem seleciona itens na Sprint Planning.

☐ Sei dividir um item grande em fatias utilizáveis e reconhecer divisões erradas por camada, fase ou especialidade.

☐ Sei onde ficam dívida técnica, defeitos, pesquisa e requisitos transversais.

☐ Sei diferenciar critérios de aceitação de Definição de Pronto.

☐ Sei o que pode e o que não pode mudar durante a Sprint e quem decide cada mudança.

☐ Sei reconhecer práticas complementares (histórias, pontos, MoSCoW, DoR, 10%) tratadas como obrigatórias.

☐ Sei como organizar backlog e Product Owner com vários Scrum Teams ou vários produtos.

Questões de revisão do capítulo

R1.1 · Gestão do Product Backlog

O Product Owner tem dois itens de valor parecido, e um deles reduz um grande risco técnico. Qual ordem tende a ser mais adequada?

A. Antecipar o item que reduz o risco, para aprender mais cedo
B. Ordenar pelos itens de menor esforço, desconsiderando o risco
C. Adiar o item de risco até que os outros itens estejam concluídos
D. Deixar a ordem para o Scrum Master, que é responsável pelos riscos

R1.2 · Gestão do Product Backlog

Dois Scrum Teams trabalham no mesmo produto e há dependências entre os itens. Onde essas dependências devem ser tratadas?

A. Em um comitê de arquitetura que define a ordem dos itens de cada time
B. Em Product Backlogs separados, sincronizados pelos Scrum Masters
C. Na Sprint Retrospective, quando os times avaliam o que atrasou
D. No refinamento do único Product Backlog, com os dois times e o Product Owner

R1.3 · Gestão do Product Backlog

Vários stakeholders têm acesso para editar o Product Backlog e incluem itens diretamente, mudando a ordem sem consultar ninguém. Qual é o risco principal?

A. A Definição de Pronto deixa de ser aplicada a esses itens
B. A ordem deixa de refletir as decisões de valor do Product Owner
C. O Scrum Master perde o controle das tarefas da Sprint
D. Os Developers passam a ter mais itens à disposição para escolher na Sprint Planning

R1.4 · Gestão do Product Backlog

Qual afirmação sobre as estimativas dos itens do Product Backlog está correta?

A. Devem ser feitas em horas e aprovadas pelo Product Owner
B. São previsões dos Developers e podem mudar conforme se aprende
C. São compromissos que não podem ser alterados após a Sprint Planning
D. São definidas pelo Scrum Master para garantir consistência

Respostas comentadas

R1.1. A) Antecipar o item que reduz o risco, para aprender mais cedo

Por quê: O Product Owner ordena o Product Backlog para maximizar valor, considerando também risco, dependências e aprendizado. Reduzir riscos cedo protege o valor futuro.

✗ B) Incorreta. Esforço é um fator, mas desconsiderar o risco pode comprometer o valor futuro.
✗ C) Incorreta. Adiar riscos aumenta o custo de descobrir problemas tarde.
✗ D) Incorreta. O Scrum Master não ordena o Product Backlog nem responde pelos riscos do produto.

> **Dica de prova:** Valor parecido? Risco e aprendizado desempatam. Antecipar incertezas costuma ser a melhor escolha.

R1.2. D) No refinamento do único Product Backlog, com os dois times e o Product Owner

Por quê: Vários Scrum Teams no mesmo produto compartilham a mesma Meta do Produto, o mesmo Product Backlog e o mesmo Product Owner. O refinamento conjunto torna as dependências transparentes.

✗ A) Incorreta. A ordem do Product Backlog é decisão do Product Owner, não de um comitê.

✗ B) Incorreta. Um produto tem um único Product Backlog. Backlogs separados escondem dependências.

✗ C) Incorreta. A Retrospective trata da forma de trabalhar. Dependências entre itens devem ser tratadas antes, no refinamento e no planejamento.

> **Dica de prova:** Dependências aparecem e são tratadas no Product Backlog único, durante o refinamento.

R1.3. B) A ordem deixa de refletir as decisões de valor do Product Owner

Por quê: Stakeholders podem sugerir itens e tentar convencer o Product Owner, mas a ordem do Product Backlog é decisão dele. Quando todos mudam a ordem, ninguém responde pelo valor.

✗ A) Incorreta. A Definição de Pronto continua valendo para qualquer trabalho que se torne Incremento.

✗ C) Incorreta. O Scrum Master não controla tarefas da Sprint.

✗ D) Incorreta. Mais itens não é o problema central. O risco é perder a coerência de valor na ordem.

> **Dica de prova:** Muitos editores no Product Backlog sem acordo com o Product Owner = responsabilidade diluída.

R1.4. B) São previsões dos Developers e podem mudar conforme se aprende

Por quê: Os Developers que farão o trabalho são responsáveis pelo dimensionamento. Estimativas são previsões e se ajustam conforme o refinamento e a experiência aumentam a clareza.

✗ A) Incorreta. O Scrum não define unidade de estimativa, e o Product Owner não aprova estimativas.

✗ C) Incorreta. Estimativas não são compromissos. No Scrum, os compromissos são a Meta do Produto, a Meta da Sprint e a Definição de Pronto.

✗ D) Incorreta. O Scrum Master não estima os itens.

> **Dica de prova:** Estimativa é previsão, não compromisso. Os compromissos do Scrum são as metas e a Definição de Pronto.

02

CAPÍTULO 2

O Framework Scrum

Este é o domínio de maior peso na distribuição de referência (a Scrum org não publica pesos por área de foco) e cobra o Scrum Guide 2020 ao pé da letra: empirismo e pensamento enxuto, pilares, valores, Scrum Team, eventos, artefatos e compromissos. Preciso saber de cor os timeboxes, quem participa de cada evento, o que cada evento inspeciona e o que sai dele. A prova adora cenários de organização querendo cortar eventos, Sprint de preparação, subtimes, Scrum Master como chefe e Daily Scrum como reunião de status. Onde mais erro: misturar pilares com valores, confundir Review com Retrospective e esquecer que a Daily Scrum não encolhe com Sprints curtas.

Neste capítulo

17 tópicos-chave · 14 pegadinhas · 7 macetes · 14 gatilhos · 4 questões comentadas

Tópicos-chave

Empirismo e pensamento enxuto

O Scrum se baseia no empirismo (conhecimento vem da experiência e de decisões baseadas no que é observado) e no pensamento enxuto (reduzir desperdício e focar no essencial). É indicado para problemas complexos, em que o que vai acontecer é incerto. Usa abordagem iterativa e incremental para otimizar a previsibilidade e controlar o risco.

Pilares do empirismo

Transparência, Inspeção e Adaptação, nessa dependência: sem transparência a inspeção é enganosa e gera desperdício. A transparência vale para quem faz e para quem recebe o trabalho. Adaptação deve ocorrer o quanto antes quando um desvio é detectado, sem esperar o próximo evento ou aprovação; é mais difícil quando as pessoas não têm autonomia.

Valores do Scrum

Compromisso, Foco, Abertura, Respeito e Coragem. Quando incorporados, os pilares ganham vida e constroem confiança; os membros aprendem os valores na prática, trabalhando com eventos e artefatos. Colaboração, previsão e transparência não são valores.

Scrum Team

Um Scrum Master, um Product Owner e Developers, sem subtimes nem hierarquias, focados em um objetivo por vez (a Meta do Produto). É multifuncional e auto-gerenciável: decide internamente quem faz o quê, quando e como. Normalmente 10 pessoas ou menos; se crescer demais, reorganiza-se em vários Scrum Teams coesos no mesmo produto.

Sprint

Contêiner de todos os eventos, com duração fixa de um mês ou menos, nunca estendida nem encurtada. Uma nova Sprint começa imediatamente após a anterior: não há intervalo nem Sprint de preparação. Só pode ser cancelada pelo Product Owner, quando a Meta da Sprint se torna obsoleta.

Sprint Planning

Inicia a Sprint e responde por que a Sprint é valiosa, o que pode ser feito e como. Participa todo o Scrum Team, que pode convidar pessoas para orientar. A Meta da Sprint precisa estar definida antes do fim; os Developers decompõem o trabalho em unidades menores, muitas vezes de um dia ou menos, e o plano evolui durante a Sprint.

Daily Scrum

Evento de 15 minutos dos Developers para inspecionar o progresso rumo à Meta da Sprint e adaptar o Sprint Backlog, gerando um plano acionável para o próximo dia. Mesmo horário e local para reduzir a complexidade. Formato livre: as três perguntas e ficar em pé são opcionais; o Product Owner só participa como Developer se trabalha em itens do Sprint Backlog.

Sprint Review

Penúltimo evento: o Scrum Team apresenta os resultados aos principais stakeholders, inspeciona o Incremento e o que mudou no ambiente e decide os próximos passos. É sessão de trabalho, não demo formal, aprovação ou portão de liberação. O Product Backlog pode sair ajustado do evento.

Sprint Retrospective

Fecha a Sprint. Só o Scrum Team, incluindo o Product Owner como membro pleno, inspeciona pessoas, interações, processos, ferramentas e a Definição de Pronto. As melhorias mais úteis são tratadas o quanto antes e podem entrar no Sprint Backlog da próxima Sprint.

Artefatos e compromissos

Product Backlog com Meta do Produto, Sprint Backlog com Meta da Sprint, Incremento com Definição de Pronto. Os compromissos reforçam o empirismo e os valores, dando foco e transparência para medir progresso. Velocidade e Definição de Preparado não são compromissos.

Meta da Sprint

Único objetivo da Sprint, criado pelo Scrum Team na Sprint Planning e adicionado ao Sprint Backlog; é compromisso dos Developers. Dá flexibilidade quanto ao trabalho exato para alcançá-la: meta fixa, caminho flexível. Sem Meta da Sprint a Sprint vira lista de tarefas.

Sprint Backlog

Composto pela Meta da Sprint (por quê), pelos itens selecionados (o quê) e pelo plano de ação para entregar o Incremento (como). É um plano feito pelos Developers e para os Developers, atualizado durante a Sprint conforme aprendem. Não inclui lista de impedimentos nem a Definição de Pronto.

Incremento

Passo concreto rumo à Meta do Produto, utilizável, aditivo a todos os anteriores e verificado para que todos funcionem juntos. Nasce no momento em que um item atende à Definição de Pronto; podem surgir vários na mesma Sprint e podem ser entregues antes da Sprint Review.

Definição de Pronto

Descrição formal do estado do Incremento quando atende às medidas de qualidade exigidas; cria entendimento comum do que está concluído. Se a organização tem padrão, ele é o mínimo; se não tem, o Scrum Team cria a sua. Vários times no mesmo produto definem e seguem a mesma; problemas de qualidade recorrentes costumam ser resolvidos fortalecendo-a.

Scrum Master

Responsável por estabelecer o Scrum e pela efetividade do Scrum Team; líder que serve ao Scrum Team, ao Product Owner e à organização. Garante que os eventos aconteçam e sejam positivos, produtivos e dentro do timebox, mas não precisa facilitar todos. Provoca a remoção de impedimentos, não distribui tarefas, não avalia desempenho individual e não decide sobre o produto.

Developers

Pessoas comprometidas em criar um Incremento utilizável a cada Sprint, incluindo testadores, designers e analistas. Criam o plano da Sprint, incorporam qualidade aderindo à Definição de Pronto, adaptam o plano diariamente e se responsabilizam mutuamente como profissionais. Colaboram além da especialidade quando a Meta da Sprint exige.

Framework intencionalmente incompleto

O Scrum define só o essencial e as regras orientam relações e interações; práticas complementares (histórias de usuário, Kanban, burn-up) podem ser acrescentadas. Implementar só partes é possível, mas o resultado não é Scrum: não se removem eventos, artefatos, compromissos ou responsabilidades. Serve para produtos que não são software.

Macetes para memorizar

Timeboxes (Sprint de um mês)

8-4-3-15: Planning, Review, Retrospective e Daily; só a Daily não encolhe.

Pilares

TIA: Transparência, Inspeção, Adaptação, em fila: sem T não há I, sem I não há A.

Valores

CFARC: Compromisso, Foco, Abertura, Respeito, Coragem.

Responsabilidades

Product Owner = valor; Scrum Master = efetividade; Developers = Incremento utilizável e qualidade.

Sprint Planning

Por quê, O quê, Como; sem Meta da Sprint a Planning não terminou.

Fim da Sprint

Review olha o produto (o quê), Retrospective olha o time (como), e a Retro apaga a luz.

Incompleto não é opcional

No Scrum pode somar práticas, nunca subtrair eventos, artefatos ou responsabilidades.

Tabela de memorização: não confunda

Quando é...	...e quando é
Pilares do empirismo Transparência, Inspeção e Adaptação: a mecânica do empirismo.	**Valores do Scrum** Compromisso, Foco, Abertura, Respeito e Coragem: comportamentos que dão vida aos pilares.
Sprint Review Inspeciona o produto (Incremento, ambiente, Product Backlog) com stakeholders; até 4 horas; penúltimo evento.	**Sprint Retrospective** Inspeciona a forma de trabalhar, só com o Scrum Team; até 3 horas; último evento.

Coragem Fazer a coisa certa sob pressão, como alertar sobre um risco sério que gera desconforto.	**Abertura** Ser transparente sobre trabalho e desafios; esconder atraso ou meta abandonada viola a Abertura.
Compromisso Empenho nas metas e apoio mútuo mesmo sob pressão.	**Respeito** Tratar todos como profissionais capazes, ouvindo o júnior com a mesma seriedade.
Cancelar a Sprint Só quando a Meta da Sprint fica obsoleta, por decisão do Product Owner.	**Renegociar escopo** Meta ainda válida, mas não dá para concluir tudo, alguém doente, item mais complexo ou pedido novo.
Scrum Master garante os eventos Correto: garante que aconteçam, sejam produtivos e dentro do timebox.	**Scrum Master facilita todos os eventos** Falso: não precisa conduzir pessoalmente; a Daily Scrum, por exemplo, é dos Developers.
Daily Scrum Developers inspecionam o progresso rumo à Meta da Sprint e replanejam o dia; cria foco e melhora o auto-gerenciamento.	**Reunião de status** Relatar tarefas ao Scrum Master, ao Product Owner ou a stakeholders: sempre alternativa errada.

Números que caem

Item	Valor
Duração máxima da Sprint	**Um mês (duração fixa, sem extensão)**
Sprint Planning (Sprint de um mês)	**Até 8 horas**
Sprint Review (Sprint de um mês)	**Até 4 horas**
Sprint Retrospective (Sprint de um mês)	**Até 3 horas**
Daily Scrum	**15 minutos, em qualquer duração de Sprint**
Tamanho do Scrum Team	**Normalmente 10 pessoas ou menos, contando Product Owner, Scrum Master e Developers**
Eventos do Scrum	**5: Sprint, Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective**
Pilares / valores	**3 pilares e 5 valores**
Artefatos / compromissos	**3 artefatos, 3 compromissos**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Respeito, colaboração ou previsão aparecem entre os pilares do empirismo.
Como não cair: Pilares são só Transparência, Inspeção e Adaptação. Respeito é valor; colaboração e previsão não são pilares.

⚠ **Alerta de prova:** Com Sprint curta, a Daily Scrum também encolhe para 10 ou 5 minutos.
Como não cair: Daily Scrum tem sempre 15 minutos. Só Planning, Review e Retrospective costumam ter timebox menor em Sprints mais curtas.

⚠ **Alerta de prova:** Refinamento do Product Backlog e Sprint Zero aparecem entre os eventos do Scrum.
Como não cair: Os cinco eventos são Sprint, Sprint Planning, Daily Scrum, Sprint Review e Sprint Retrospective. Refinamento é atividade; Sprint Zero não existe.

⚠ **Alerta de prova:** A Sprint Planning acontece antes da Sprint, ou existe intervalo entre Sprints para planejamento.
Como não cair: A Sprint Planning inicia a Sprint e faz parte dela; a próxima Sprint começa imediatamente após a anterior.

⚠ **Alerta de prova:** Time experiente pode eliminar a Sprint Planning ou a Retrospective e continuar sendo Scrum, se o Scrum Master aprovar.
Como não cair: Ninguém autoriza remover eventos mantendo o resultado como Scrum. Maturidade torna os eventos mais eficientes, não os dispensa.

⚠ **Alerta de prova:** O Scrum Guide recomenda Sprints de duas semanas, ou a Sprint pode passar de um mês se o Product Owner aprovar.
Como não cair: O Scrum Guide só define o limite de um mês ou menos, com duração fixa, sem exceções.

⚠ **Alerta de prova:** Time de 16 pessoas: nomear segundo Product Owner, criar subtimes com líderes ou alongar a Daily Scrum.
Como não cair: Reorganizar em vários Scrum Teams coesos com a mesma Meta do Produto, o mesmo Product Backlog e o mesmo Product Owner.

⚠ **Alerta de prova:** O Scrum Guide 2020 exige as três perguntas na Daily Scrum, ou mudar o formato depende do Scrum Master ou do Product Owner.
Como não cair: As três perguntas foram removidas; os Developers escolhem estrutura e técnicas, mantendo 15 minutos e o foco na Meta da Sprint.

⚠ **Alerta de prova:** Stakeholders e gerência participam da Sprint Retrospective, ou o Product Owner fica de fora para os Developers falarem à vontade.
Como não cair: A Retrospective é do Scrum Team inteiro, com o Product Owner como membro pleno, e sem stakeholders.

⚠ **Alerta de prova:** Ajuste de processo fora dos limites aceitáveis deve esperar a próxima Retrospective.
Como não cair: O Scrum Guide pede ajuste o quanto antes para minimizar desvios; eventos são oportunidades formais, não as únicas.

⚠ **Alerta de prova:** Pular testes perto do fim da Sprint é aceitável se a dívida técnica for registrada ou se a Definição de Pronto for suspensa.
Como não cair: A qualidade não diminui durante a Sprint e a Definição de Pronto não é suspensa; o escopo é renegociado com o Product Owner.

⚠ **Alerta de prova:** Esconder atrasos dos stakeholders compromete primeiro a Inspeção ou a Adaptação.
Como não cair: O primeiro pilar atingido é a Transparência; inspeção e adaptação sofrem como consequência.

⚠ **Alerta de prova:** Concluiu tudo três dias antes: a Sprint termina antes; ou atrasou: estende-se.
Como não cair: Duração fixa: a Sprint segue até o fim e o time combina com o Product Owner o uso do tempo.

⚠ **Alerta de prova:** Scrum Master deve aplicar advertência a um Developer que descumpre combinados, ou avaliar desempenho individual.
Como não cair: Developers se responsabilizam mutuamente como profissionais; o Scrum Master faz coaching em auto-gerenciamento e não pune nem avalia indivíduos.

Como cai na prova

📝 **Como cai:** Pergunta direta de timebox, com distratores que são timeboxes de outros eventos.
Como resolver: Repita 8-4-3-15 e confira o evento pedido. Se a Sprint é mais curta, só Planning, Review e Retrospective encolhem; a Daily fica em 15 minutos.

📝 **Como cai:** Comportamento descrito e pergunta qual valor ou pilar fica evidente ou comprometido.
Como resolver: Primeiro decida se pede pilar ou valor. Informação escondida: pilar Transparência, valor Abertura. Sob pressão fazendo o certo: Coragem. Empenho e apoio: Compromisso. Tratar como capaz: Respeito.

📝 **Como cai:** Organização ou time quer adaptar o Scrum: cortar evento, pausa entre Sprints, subtimes, Sprint maior.
Como resolver: Pode acrescentar práticas, nunca remover elementos. Marque a alternativa em que o Scrum Master ajuda a entender o propósito de cada elemento.

📝 **Como cai:** Escolha DUAS ou TRÊS sobre eventos, artefatos, compromissos ou Sprint Backlog.
Como resolver: Use as listas fixas: 5 eventos, 3 artefatos, 3 compromissos, Sprint Backlog = meta + itens + plano. Elimine refinamento, Sprint Zero, velocidade, Definição de Preparado e lista de impedimentos.

📝 **Como cai:** Quem participa ou conduz um evento (Product Owner na Daily, stakeholder na Retrospective).
Como resolver: Daily: Developers. Review: Scrum Team e stakeholders. Retrospective e Planning: Scrum Team (Planning aceita convidados para orientar). Product Owner na Daily só como Developer.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
Sprint de um mês: timebox de Planning, Review ou Retrospective	**8, 4 e 3 horas; Daily Scrum sempre 15 minutos.**
Decisões baseadas no que é observado, problema complexo e incerto	**Empirismo (e pensamento enxuto: menos desperdício, foco no essencial).**
Informação escondida ou maquiada	**Pilar Transparência; valor Abertura.**
Alertar risco sério sob pressão do patrocinador	**Coragem.**
Ouvir o júnior com a mesma seriedade	**Respeito.**
Concentrar-se no trabalho da Sprint, poucas frentes ao mesmo tempo	**Foco.**
Meta da Sprint obsoleta (mudança regulatória, mercado)	**Product Owner pode cancelar a Sprint.**
Subtime de testes, líder técnico que distribui tarefas	**Contraria o Scrum: sem subtimes nem hierarquias; Developers se auto-gerenciam.**
Organização mede horas, quer cortar eventos ou acha que a Review é aprovação formal	**Scrum Master ajuda a organização a entender o Scrum e o empirismo.**
Bloqueio que depende de outro departamento	**Scrum Master provoca a remoção de impedimentos.**
Defeitos chegam com frequência aos usuários	**Fortalecer a Definição de Pronto na Retrospective.**
Quando nasce o Incremento?	**Quando um item atende à Definição de Pronto, a qualquer momento da Sprint.**

Por que mesmo horário e local na Daily Scrum?	**Para reduzir a complexidade.**
Novo time sem Definição de Pronto da empresa	**O próprio Scrum Team cria a sua, adequada ao produto.**

🎯 Se você lembrar de só uma coisa deste capítulo...

- 8-4-3-15: só a Daily Scrum não encolhe com Sprints mais curtas.
- Só o Product Owner cancela a Sprint, e só quando a Meta da Sprint fica obsoleta.
- Sem subtimes, sem hierarquias, sem Sprint Zero e sem intervalo entre Sprints.
- Incremento nasce ao atender à Definição de Pronto e pode ser entregue antes da Sprint Review.
- Scrum Master responde pela efetividade, garante os eventos, mas não precisa conduzir todos.

Essenciais para a prova (checklist)

☐ Sei os timeboxes de todos os eventos para uma Sprint de um mês e o que muda em Sprints curtas.
☐ Sei separar os 3 pilares dos 5 valores e associar cada valor a um comportamento.
☐ Sei o propósito, os participantes e o resultado de cada um dos 5 eventos.
☐ Sei os 3 artefatos com seus compromissos e a composição do Sprint Backlog.
☐ Sei quando uma Sprint pode ser cancelada e por quem.
☐ Sei as responsabilidades de Product Owner, Scrum Master e Developers e os serviços do Scrum Master aos três públicos.
☐ Sei quem cria a Definição de Pronto com e sem padrão da organização.
☐ Sei por que remover eventos ou artefatos faz o resultado deixar de ser Scrum.
☐ Sei o que fazer quando o Scrum Team fica grande demais.

Questões de revisão do capítulo

R2.1 · O Framework Scrum

A Sprint Retrospective do time virou uma lista de reclamações sem ações. Qual é o resultado esperado do evento?

A. Revisar o Incremento com os stakeholders
B. Definir as melhorias mais úteis e agir sobre elas
C. Registrar as reclamações e enviá-las à gerência para análise e decisão
D. Avaliar o desempenho de cada membro do time

R2.2 · O Framework Scrum

Quem participa da Sprint Retrospective?

A. O Scrum Team e os principais stakeholders
B. O Scrum Master e a gerência da área
C. O Scrum Team, com o Product Owner
D. Somente os Developers e o Scrum Master

R2.3 · O Framework Scrum

Quem participa da Sprint Planning?

A. O Product Owner e o Scrum Master, sem os Developers
B. O Scrum Master e um representante da gerência
C. Os Developers e os stakeholders, sem o Product Owner
D. Todo o Scrum Team, podendo convidar outras pessoas para orientar

R2.4 · O Framework Scrum

Um Developer descumpre com frequência os combinados do time. Segundo o Scrum Guide, qual responsabilidade dos Developers se aplica a essa situação?

A. Assumir responsabilidade mútua como profissionais
B. Aguardar a avaliação anual do gerente de linha
C. Pedir ao Scrum Master que aplique uma advertência
D. Reportar o colega ao Product Owner para avaliação

Respostas comentadas

R2.1. B) Definir as melhorias mais úteis e agir sobre elas

Por quê: O propósito da Sprint Retrospective é planejar formas de aumentar a qualidade e a efetividade. O time identifica as mudanças mais úteis, e as mais impactantes são tratadas o quanto antes.

✗ A) Incorreta. Revisar o Incremento com stakeholders é o propósito da Sprint Review.
✗ C) Incorreta. O Scrum Team é auto-gerenciável e age sobre suas melhorias.
✗ D) Incorreta. A Retrospective não avalia desempenho individual.

> **Dica de prova:** Retrospective sem ação não cumpre o propósito. Saia com melhorias concretas.

R2.2. C) O Scrum Team, com o Product Owner

Por quê: A Sprint Retrospective é o evento em que o Scrum Team inspeciona como foi a Sprint e planeja melhorias. O Scrum Team é formado por Product Owner, Scrum Master e Developers.

✗ A) Incorreta. Stakeholders colaboram na Sprint Review. A Sprint Retrospective é do Scrum Team.
✗ B) Incorreta. A gerência não participa da Sprint Retrospective.
✗ D) Incorreta. O Product Owner é membro do Scrum Team e participa da Sprint Retrospective.

> **Dica de prova:** Retrospective: Scrum Team completo, sem stakeholders.

R2.3. D) Todo o Scrum Team, podendo convidar outras pessoas para orientar

Por quê: A Sprint Planning inicia a Sprint definindo o trabalho a ser feito. O plano resultante é criado pelo trabalho colaborativo de todo o Scrum Team, que pode convidar outras pessoas para orientar.

✗ A) Incorreta. Os Developers são essenciais: eles selecionam os itens e planejam o trabalho.
✗ B) Incorreta. A Sprint Planning é do Scrum Team, não da gerência.
✗ C) Incorreta. O Product Owner participa e propõe como o produto pode aumentar seu valor.

> **Dica de prova:** Sprint Planning: Scrum Team completo, com convidados opcionais.

R2.4. A) Assumir responsabilidade mútua como profissionais

Por quê: Os Developers são responsáveis por criar o plano da Sprint, incorporar qualidade, adaptar o plano diariamente e responsabilizar-se mutuamente como profissionais.

✗ B) Incorreta. Esperar uma avaliação anual adia a solução e ignora o auto-gerenciamento do time.
✗ C) Incorreta. O Scrum Master não aplica punições. Ele pode ajudar o time com coaching em auto-gerenciamento.
✗ D) Incorreta. O Product Owner não avalia o desempenho dos Developers.

> **Dica de prova:** No auto-gerenciamento, os próprios Developers tratam os combinados entre si, com apoio do Scrum Master se necessário.

03

CAPÍTULO 3

Responsabilidades do Product Owner

Aqui eu preciso saber exatamente o que é do Product Owner e o que não é: ele responde por maximizar o valor do produto, gerencia o Product Backlog e decide ordem, conteúdo, cancelamento da Sprint e liberação. A prova testa muito a fronteira com os Developers (o como, o quem, as estimativas, o Sprint Backlog) e com a organização (comitês, diretores, autoridade). O tema mais repetido é delegar trabalho sem delegar responsabilidade, e o Product Owner como uma pessoa, não um comitê. Onde mais erro: achar que o Product Owner define sozinho a Meta da Sprint ou a Definição de Pronto, ou que pode impor itens ao Sprint Backlog.

Neste capítulo
17 tópicos-chave · 13 pegadinhas · 5 macetes · 13 gatilhos · 4 questões comentadas

Tópicos-chave

Accountability do Product Owner

Maximizar o valor do produto resultante do trabalho do Scrum Team. Também é responsável pela gestão efetiva do Product Backlog. Efetividade do time é do Scrum Master; Incremento utilizável e qualidade, dos Developers.

Gestão efetiva do Product Backlog

Quatro atividades: desenvolver e comunicar explicitamente a Meta do Produto, criar e comunicar claramente os itens, ordenar os itens e garantir que o Product Backlog seja transparente, visível e compreendido.

Delegar trabalho, não responsabilidade

O Product Owner pode delegar a outras pessoas trabalho como escrever e detalhar itens ou organizar informações para o refinamento, mas continua responsável pela ordem do Product Backlog e pela Meta do Produto. Se delega e deixa de acompanhar, responde mesmo assim pelo resultado. Em férias, delega com limites de decisão combinados.

Uma pessoa, não um comitê

O Product Owner é uma pessoa para haver responsabilidade clara pelo valor. Um comitê pode representar necessidades ou aconselhar, mas não substituí-lo; a decisão e a responsabilidade continuam dele. Produto grande com vários times continua com um Product Owner, que pode delegar trabalho.

Autoridade e respeito da organização

Para o Product Owner ter sucesso, a organização inteira precisa respeitar suas decisões. Quem quer mudar o Product Backlog tenta convencê-lo; gerente que altera a ordem diretamente contraria o Scrum Guide. Sinais de autoridade real: pode dizer não e suas decisões sobre a ordem são respeitadas.

Decisões que cabem ao Product Owner

Ordem e conteúdo do Product Backlog, cancelamento da Sprint quando a Meta da Sprint fica obsoleta e liberação de um Incremento pronto (decisão de valor para o negócio). As decisões ficam visíveis no conteúdo e na ordem do Product Backlog e no Incremento inspecionável na Sprint Review.

O que não cabe ao Product Owner

Definir quem faz cada tarefa, impor estimativas, escolher tecnologia e arquitetura (o como é dos Developers), alterar sozinho a Definição de Pronto ou impor itens ao Sprint Backlog. Ele esclarece e ordena; os Developers organizam e executam.

Product Owner na Sprint Planning

Prepara os participantes para discutir os itens mais importantes e propõe como o produto pode aumentar seu valor e utilidade na Sprint. Confiar nos Developers não o dispensa do evento. A Meta da Sprint é definida pelo Scrum Team inteiro: se os Developers acham inviável, ajusta-se em conjunto.

Previsão x compromisso na Sprint

Os itens selecionados são previsão; o compromisso é a Meta da Sprint. Exigir compromisso com todos os itens remove a flexibilidade. Capacidade é dos Developers; o Product Owner respeita a previsão e ajuda a escolher o que traz mais valor.

Product Owner durante a Sprint

Fica disponível para esclarecer itens e negociar detalhes e escopo, protegendo a Meta da Sprint. Pedido novo sem relação com a meta vai para o Product Backlog; item que perdeu valor com meta válida é retirado em conversa com os Developers, sem cancelar. Ausência frequente atrasa dúvidas e reduz o valor, sem transferir a responsabilidade.

Product Owner como Developer

Pode trabalhar em itens do Sprint Backlog; nesse caso participa da Daily Scrum como Developer, nunca para cobrar ou redistribuir tarefas.

Definição de Pronto e o Product Owner

A Definição de Pronto vem do padrão da organização ou é criada pelo Scrum Team; o Product Owner propõe mudanças como membro do time, não por decreto. Perto de uma data importante, não pode pedir para pular critérios: negocia-se escopo.

Pressão de prazo e preço fixo

Com qualidade caindo por pressão de prazo, ouve os Developers, preserva a Definição de Pronto e ajusta escopo e expectativas. Em contrato de preço e prazo fixos, o escopo é tratado como variável, priorizado pelo valor.

Tomador de pedidos

Disfunção em que o Product Owner registra o que outros decidem, aceita pedidos sem avaliar valor ou deixa a ordem para quem pede mais alto. Dizer não com transparência faz parte do papel, pois recursos são limitados e nem todo pedido contribui para a Meta do Produto.

Proximidade de clientes e mercado

O Product Owner precisa conhecer clientes, usuários, mercado e estratégia; escolhido só pelo conhecimento técnico, tem dificuldade em decidir valor. Longe dos clientes ou dependente de intermediários, decide com informações filtradas e aprende mais devagar. Seu tempo vale mais com clientes, dados e stakeholders do que gerindo tarefas.

Colaboração com os Developers

Explica o porquê e o problema e deixa os Developers definirem a solução; consulta-os sobre riscos técnicos que não domina. Diante de estimativa alta, ajuda a entender o item e escolher trade-offs. Item valioso e arriscado pede reduzir o risco primeiro, com experimento inicial.

Scrum Master apoiando o Product Owner

O Scrum Master ajuda a encontrar técnicas para definir a Meta do Produto e gerenciar o Product Backlog, a estabelecer planejamento empírico e facilita a colaboração com stakeholders quando solicitado ou necessário. Nunca ordena o backlog, não aprova a Meta do Produto e não negocia prazos em nome do time.

Macetes para memorizar

💡 Delegação

Mão de obra se delega; assinatura não.

💡 Fronteira com os Developers

Product Owner: o quê e o porquê. Developers: o como, o quem e o quanto.

Poderes do Product Owner

OCL: Ordem do backlog, Cancelamento da Sprint, Liberação do Incremento.

Pressão

Aperto de prazo corta escopo, nunca a Definição de Pronto.

Tomador de pedidos

Product Owner que diz sim a tudo não maximiza valor.

Tabela de memorização: não confunda

Quando é...	...e quando é
Delegar trabalho Permitido: analista escreve e detalha itens, organiza informações para o refinamento.	**Delegar responsabilidade** Impossível: ordem do Product Backlog e Meta do Produto continuam respondidas pelo Product Owner.
O quê e o porquê Product Owner: valor, ordem, problema a resolver.	**O como e o quem** Developers: tecnologia, arquitetura, tarefas, estimativas e plano da Sprint.
Product Backlog Do Product Owner: muda a qualquer momento por decisão dele.	**Sprint Backlog** Dos Developers: inclusão de item urgente é negociação, não imposição.
Pronto Questão técnica: atende à Definição de Pronto (Developers).	**Liberar** Decisão de valor para o negócio: Product Owner, sem esperar a Sprint Review.
Comitê que aconselha Aceitável: sugere, e o Product Owner decide e responde.	**Comitê que atua como Product Owner** Sempre errado: o Product Owner é uma pessoa.
Meta da Sprint Compromisso dos Developers, criado pelo Scrum Team.	**Itens selecionados** Previsão, que pode ser renegociada sem ameaçar a meta.

Números que caem

Item	Valor
Product Owners por produto	**1, mesmo com vários Scrum Teams**
Atividades de gestão do Product Backlog	**4: Meta do Produto, itens claros, ordem e transparência**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Ao delegar a ordenação a um analista, a responsabilidade passa a ser dele.
Como não cair: Delegou trabalho, não responsabilidade: o Product Owner continua respondendo pelo valor.

⚠ **Alerta de prova:** Comitê de três gerentes como Product Owner para dividir responsabilidade.
Como não cair: O Product Owner é uma pessoa; comitê pode representar necessidades, não substituí-lo.

⚠ **Alerta de prova:** Com cinco Scrum Teams, um Product Owner por time.
Como não cair: Escalar não multiplica Product Owners: um produto, um Product Owner, que pode delegar trabalho.

⚠ **Alerta de prova:** O Product Owner define sozinho a Meta da Sprint.
Como não cair: Ele propõe valor; a Meta da Sprint é criada pelo Scrum Team inteiro e é compromisso dos Developers.

⚠ **Alerta de prova:** O Product Owner pode alterar sozinho a Definição de Pronto ou pedir para pular critérios perto de uma data.
Como não cair: A Definição de Pronto é da organização ou do Scrum Team; ele propõe mudanças como membro, e prazo apertado reduz escopo, não qualidade.

⚠ **Alerta de prova:** Item urgente no meio da Sprint: o Product Owner o coloca direto no Sprint Backlog.
Como não cair: Conversa com os Developers, que decidem se cabe sem arriscar a Meta da Sprint; sem relação com a meta, vai para o Product Backlog.

⚠ **Alerta de prova:** Product Owner escolhe tecnologia e arquitetura porque responde pelo valor.
Como não cair: Como transformar itens em Incremento é decisão dos Developers.

⚠ **Alerta de prova:** Product Owner impõe estimativa menor ou define quem faz cada tarefa.
Como não cair: Isso fere o auto-gerenciamento; ele influencia explicando o item e oferecendo trade-offs.

⚠ **Alerta de prova:** O Product Owner pode faltar à Sprint Planning porque confia nos Developers.
Como não cair: Sem ele, a Planning perde a proposta de valor e a conversa sobre os itens.

⚠ **Alerta de prova:** Exigir que os Developers se comprometam com todos os itens selecionados.
Como não cair: Itens são previsão; compromisso é a Meta da Sprint.

⚠ **Alerta de prova:** Diretor ou gerente altera a ordem do backlog, e o Product Owner deve aceitar pela hierarquia.
Como não cair: A organização deve respeitar as decisões do Product Owner; quem quer mudar tenta convencê-lo.

⚠ **Alerta de prova:** Seguir o roadmap aprovado mesmo com dados de uso contrários, porque foi aprovado.
Como não cair: Apresentar as evidências e negociar a revisão do roadmap com transparência.

⚠ **Alerta de prova:** Avaliar o Product Owner pela velocidade do time.
Como não cair: Isso incentiva produção em vez de valor; avalie por valor e resultados.

Como cai na prova

📝 **Como cai:** Alguém de fora (diretor, gerente, comitê, patrocinador) pressiona o Product Owner ou mexe no backlog.
Como resolver: Lembre que a organização deve respeitar as decisões do Product Owner. A resposta certa é ele decidir com base em valor e evidências, explicar com transparência e levar pedidos ao Product Backlog.

📝 **Como cai:** Product Owner delega, se ausenta ou o produto cresce.
Como resolver: Separe trabalho de responsabilidade: trabalho pode ser delegado com limites; responsabilidade pela ordem e pela Meta do Produto continua com ele; nunca multiplique Product Owners.

📝 **Como cai:** Product Owner tenta decidir algo dos Developers (tarefas, estimativas, tecnologia, Sprint Backlog).
Como resolver: Marque a alternativa que devolve a decisão aos Developers e mantém o Product Owner explicando o problema, o valor e os trade-offs.

📝 **Como cai:** Escolha DUAS sobre decisões ou atitudes do Product Owner.
Como resolver: Procure ordem do backlog, cancelamento da Sprint, Meta do Produto comunicada, dizer não, medir resultados e reordenar após feedback. Descarte atribuir tarefas, impor estimativas e aceitar pedidos sem avaliar.

📝 **Como cai:** Situação durante a Sprint envolvendo pedido urgente ou item que perdeu valor.
Como resolver: Pergunte se a Meta da Sprint continua válida. Se sim, renegocie com os Developers ou mande o pedido ao Product Backlog; cancelamento só com meta obsoleta.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
Principal responsabilidade (accountability) do Product Owner	**Maximizar o valor do produto resultante do trabalho do Scrum Team.**
Analista de negócios escreve itens a pedido do Product Owner	**Permitido; responsabilidade pelo Product Backlog continua com o Product Owner.**
Quem decide liberar um Incremento pronto	**O Product Owner (decisão de valor).**
Quem pode cancelar a Sprint	**Somente o Product Owner, com Meta da Sprint obsoleta.**
Onde ficam visíveis as decisões do Product Owner	**Conteúdo e ordem do Product Backlog e Incremento inspecionável na Sprint Review.**
Dois diretores com pedidos conflitantes e prioridade máxima	**Product Owner avalia valor rumo à Meta do Produto e decide a ordem com transparência.**
Product Owner só registra o que a diretoria decide	**Sem autoridade não maximiza valor: Scrum disfuncional.**
Patrocinador quer trocar metade dos itens da Sprint, meta válida	**Meta da Sprint protegida; pedidos vão para o Product Backlog.**
Developers alertam que item do fim reduz risco técnico grave	**Considerar o alerta ao ordenar: risco afeta o valor futuro.**
Product Owner quer trabalhar em itens da Sprint	**Pode; participa da Daily Scrum como Developer.**
Ideia própria do Product Owner sem evidência	**Tratar como hipótese e testar com pouco esforço.**
Produto grande demais para a atenção do Product Owner	**Delegar parte do trabalho, mantendo a responsabilidade.**
Product Owner sem contato com clientes, só via intermediários	**Informação filtrada, menos valor, tendência a tomador de pedidos.**

🎯 Se você lembrar de só uma coisa deste capítulo...

- Delegar trabalho sim, delegar responsabilidade não.
- O Product Owner é uma pessoa, não um comitê, e é um só por produto.
- A organização inteira precisa respeitar as decisões do Product Owner.
- Meta da Sprint é do Scrum Team; Definição de Pronto é da organização ou do Scrum Team; Sprint Backlog é dos Developers.
- Cancelar a Sprint e liberar o Incremento são decisões do Product Owner.

Essenciais para a prova (checklist)

- ☐ Sei a accountability do Product Owner e as quatro atividades de gestão do Product Backlog.
- ☐ Sei o que o Product Owner pode delegar e o que continua sendo dele.
- ☐ Sei por que o Product Owner é uma pessoa e não um comitê, mesmo com vários times.
- ☐ Sei listar as decisões que cabem ao Product Owner e as que cabem aos Developers.
- ☐ Sei o papel do Product Owner na Sprint Planning, durante a Sprint e na Sprint Review.
- ☐ Sei responder quando um gerente ou diretor tenta mudar a ordem do backlog.
- ☐ Sei reconhecer o Product Owner tomador de pedidos e o que fazer.
- ☐ Sei como o Product Owner influencia a Definição de Pronto e lida com pressão de prazo.
- ☐ Sei como o Scrum Master apoia o Product Owner sem decidir por ele.

Questões de revisão do capítulo

R3.1 · Responsabilidades do Product Owner

Um novo Product Owner quer entender como as decisões dele ficam visíveis para a organização. Segundo o Scrum Guide, onde elas aparecem?

A. Nas atas de reunião aprovadas pelo comitê de produto a cada trimestre
B. Nos relatórios de velocidade que o Scrum Master envia aos stakeholders
C. No Sprint Backlog, que registra as decisões de negócio tomadas ao longo da Sprint
D. No conteúdo e na ordem do Product Backlog e no Incremento da Sprint Review

R3.2 · Responsabilidades do Product Owner

Um cliente quer contratar o desenvolvimento com preço e prazo fixos. Como o Product Owner pode manter o empirismo?

A. Pedir que o Scrum Master negocie cada mudança de escopo
B. Recusar contratos com preço fixo, pois o Scrum os proíbe
C. Fixar também o escopo e cortar testes se o prazo apertar
D. Tratar o escopo como variável, priorizado pelo valor

R3.3 · Responsabilidades do Product Owner

Os Developers avisam que a qualidade técnica está caindo por causa da pressão de prazo. O que o Product Owner deve fazer?

A. Ouvir os Developers, preservar a Definição de Pronto e ajustar escopo e expectativas
B. Pedir que os Developers mantenham o ritmo e corrijam a qualidade depois
C. Reduzir temporariamente a Definição de Pronto até a data de entrega
D. Encaminhar o assunto ao Scrum Master, pois qualidade não é tema de produto

R3.4 · Responsabilidades do Product Owner

O Product Owner também quer trabalhar em alguns itens do Sprint Backlog. O que o Scrum Guide indica?

A. Não é permitido sem autorização prévia do Scrum Master
B. É possível; nesse caso ele participa da Daily Scrum como Developer
C. Não é permitido; o Product Owner não pode fazer trabalho da Sprint
D. É possível, mas ele deixa de ser responsável pelo Product Backlog

Respostas comentadas

R3.1. D) No conteúdo e na ordem do Product Backlog e no Incremento da Sprint Review

Por quê: Para que o Product Owner tenha sucesso, a organização deve respeitar suas decisões. Elas ficam visíveis no conteúdo e na ordem do Product Backlog e no Incremento inspecionável na Sprint Review.

✗ A) Incorreta. O Scrum Guide não usa atas de comitê para dar visibilidade às decisões do Product Owner.
✗ B) Incorreta. Relatórios de velocidade não fazem parte do Scrum e não mostram as decisões de valor do Product Owner.
✗ C) Incorreta. O Sprint Backlog é o plano dos Developers para a Sprint, não um registro das decisões de negócio do Product Owner.

> **Dica de prova:** Onde ver as decisões do Product Owner: Product Backlog (conteúdo e ordem) e Incremento na Sprint Review.

R3.2. D) Tratar o escopo como variável, priorizado pelo valor

Por quê: Em trabalho complexo, fixar preço, prazo e escopo ao mesmo tempo não funciona. O Product Owner mantém o escopo flexível e ordenado por valor, com Incrementos a cada Sprint.

✗ A) Incorreta. Negociar escopo com o cliente é papel do Product Owner.
✗ B) Incorreta. O Scrum não proíbe contratos com preço fixo.
✗ C) Incorreta. Fixar tudo e cortar testes reduz a qualidade, que não diminui no Scrum.

> **Dica de prova:** Preço e prazo fixos: escopo variável, ordenado por valor. Qualidade não entra no corte.

R3.3. A) Ouvir os Developers, preservar a Definição de Pronto e ajustar escopo e expectativas

Por quê: A qualidade não diminui no Scrum. A Definição de Pronto protege a qualidade, e o Product Owner equilibra escopo e expectativas para maximizar valor de forma sustentável.

✗ B) Incorreta. Adiar a qualidade gera dívida técnica e reduz a capacidade de entregar valor no futuro.
✗ C) Incorreta. A qualidade não diminui durante a Sprint e a Definição de Pronto não é reduzida por pressão de prazo.
✗ D) Incorreta. Qualidade afeta diretamente o valor do produto e é tema do Product Owner.

> **Dica de prova:** Pressão de prazo: negocie escopo. A qualidade da Definição de Pronto não entra na negociação.

R3.4. B) É possível; nesse caso ele participa da Daily Scrum como Developer

Por quê: A Daily Scrum é para os Developers. Se o Product Owner ou o Scrum Master estiverem trabalhando ativamente em itens do Sprint Backlog, participam como Developers, sem perder suas responsabilidades.

✗ A) Incorreta. O Scrum Master não autoriza essa situação. O Scrum Team se auto-gerencia.

✗ C) Incorreta. O Scrum Guide prevê a situação em que o Product Owner trabalha em itens do Sprint Backlog.

✗ D) Incorreta. Trabalhar em itens não altera a responsabilidade do Product Owner pelo Product Backlog.

> **Dica de prova:** Trabalhar em itens do Sprint Backlog é possível para o Product Owner, que então atua também como Developer.

04

CAPÍTULO 4

Stakeholders, Previsão e Entregas

Neste domínio eu preciso saber como o Product Owner colabora com stakeholders, como prevê entregas em trabalho complexo e como o Incremento chega aos usuários. A prova cobra a Sprint Review como sessão de trabalho com stakeholders, o limite de onde eles participam, previsões com dados (burn-down, burn-up, fluxo cumulativo) e o tratamento de datas fixas. Também cai muito liberação antes do fim da Sprint, vários times com Incremento integrado e restrições da organização que atrasam valor. Onde mais erro: aceitar garantias de data, comparar velocidade entre times e achar que a Sprint Review é portão de liberação.

Neste capítulo
18 tópicos-chave · 13 pegadinhas · 6 macetes · 13 gatilhos · 4 questões comentadas

Tópicos-chave

Stakeholders na Sprint Review

O Scrum Team e os stakeholders relevantes colaboram sobre o que foi feito, o que mudou no ambiente e o que fazer a seguir; o Product Backlog pode sair ajustado. Review sem stakeholders vira demo interna e perde o feedback. Só se apresenta o que atende à Definição de Pronto.

Limites de participação

Stakeholders participam da Sprint Review, não da Daily Scrum (dos Developers) nem da Sprint Retrospective (do Scrum Team). Feedback deles cabe em outros momentos. Quem não pode ir à Review tem o feedback buscado por outros meios pelo Product Owner; eventos não são adiados por ausência.

Influenciar o Product Backlog

Quem quer mudar a ordem precisa convencer o Product Owner, que equilibra as necessidades dos stakeholders. Feedback contraditório é avaliado e decidido por ele, com transparência. Feedback que contraria a visão é analisado frente à visão e às evidências, sem aceitar ou rejeitar automaticamente.

Práticas de previsão

O Scrum Guide 2020 cita burn-downs, burn-ups e fluxos cumulativos como úteis para prever progresso, mas eles não substituem o empirismo. Previsão não é compromisso nem garantia.

Burn-up com linha de escopo

Mostra o trabalho concluído ao longo do tempo em relação ao escopo total previsto. Com a linha de escopo, torna visível quando o escopo aumenta a cada Sprint. É bom para conversas com stakeholders.

Diagrama de fluxo cumulativo

Mostra onde o trabalho se acumula no fluxo do time, revelando gargalos. Burn-up mostra progresso e escopo; o fluxo cumulativo mostra gargalos.

Previsão em faixa e probabilística

Use o histórico do time, comunique uma faixa de datas ou previsão probabilística e revise a cada Sprint. Exemplo: 8 a 12 itens por Sprint e cerca de 40 itens restantes dão entre quatro e cinco Sprints. Número exato em trabalho complexo é falsa certeza; previsão que muda com dados novos é transparência, não falha.

Velocidade

Serve para o próprio time prever, não para a gestão comparar times, porque cada time estima de forma própria. Burn-down da Sprint é usado pelos Developers, se útil, para acompanhar a Meta da Sprint, não para controle externo.

Data fixa e trade-offs

Com data legal, campanha ou preço e prazo fixos, a data vira restrição transparente e o escopo varia, priorizando o essencial. Escopo, data e valor se negociam; a qualidade definida na Definição de Pronto e a duração da Sprint não. Mais escopo sem mudar a data pede conversa com dados sobre trade-offs.

Liberação a qualquer momento

Incremento que atende à Definição de Pronto pode ser entregue aos stakeholders antes do fim da Sprint; a Sprint Review não é portão de liberação. Vários Incrementos podem ser criados e entregues na Sprint, inclusive várias implantações por dia. A Review pode inspecionar algo já em produção.

Liberação gradual e funcionalidade desligada

Para reduzir risco ou obter feedback real antes de todos, libere para parte dos usuários, meça e amplie. Liberar desligado e ativar aos poucos separa liberar de ativar, reduzindo risco sem atrasar entregas.

Vários Scrum Teams no mesmo produto

Ao fim de cada Sprint deve existir um Incremento integrado que atenda à Definição de Pronto comum; os times definem juntos e seguem a mesma Definição de Pronto. A Sprint Review inspeciona o Incremento integrado do produto com os stakeholders.

Conformidade e qualidade na Definição de Pronto

Critérios regulatórios, de segurança ou de desempenho entram na Definição de Pronto, não em fases posteriores. Se falta um requisito de qualidade importante (como testes de desempenho antes de picos de acesso), fortalece-se a Definição de Pronto com o time.

Restrições organizacionais

Comitê de mudanças que atrasa semanas, janelas trimestrais ou liberação duas vezes por ano deixam valor e feedback parados. O papel do Product Owner é tornar visível o custo desse atraso e tratar com a organização, não contornar nem aceitar calado.

Más notícias cedo

Se a Meta do Produto não será alcançada na data ou uma entrega combinada não é possível, informe cedo, com dados e alternativas, nunca à custa da qualidade. Meta do Produto abandonada é comunicada assim que a decisão é tomada.

Item que não atende à Definição de Pronto

Não pode ser liberado nem apresentado na Sprint Review e volta ao Product Backlog. Não existe item parcialmente pronto.

Transparência no lugar de relatórios

Pedido de relatório semanal ou de acompanhamento constante se responde com transparência dos artefatos: Product Backlog visível, progresso rumo à meta e Incremento. Stakeholder pode ter acesso ao Product Backlog em tempo real; progresso não se mostra com horas ou status individual.

Custo fixo por Sprint

Com custo praticamente fixo por Sprint, o Product Owner relaciona custo, tempo e valor entregue: a pergunta passa a ser se o valor de cada Sprint compensa. Sprints de duração fixa e Incrementos frequentes também aumentam a previsibilidade.

Macetes para memorizar

💡 Onde vai o stakeholder

Review sim; Daily e Retrospective não.

💡 Previsão

Histórico + faixa + revisão a cada Sprint = previsão honesta; data única = falsa certeza.

💡 Triângulo de negociação

Escopo, data e valor se negociam; qualidade (Definição de Pronto) nunca.

💡 Gráficos

Burn-up sobe e mostra o teto mexendo (escopo); fluxo cumulativo mostra o engarrafamento (gargalo).

Liberar

Pronto pela Definição de Pronto + decisão de valor do Product Owner = pode liberar, sem esperar a Review.

Más notícias

Cedo, com dados e com alternativas.

Tabela de memorização: não confunda

Quando é...	...e quando é
Previsão Baseada no histórico, em faixa, atualizada a cada Sprint: a resposta certa a patrocinadores.	**Garantia de data** Sempre alternativa errada em trabalho complexo; prometer datas sem os Developers gera pressão sobre a qualidade.
Burn-up Progresso e mudanças de escopo no mesmo gráfico.	**Fluxo cumulativo** Onde o trabalho se acumula: gargalos no fluxo.
Sprint Review Inspeção e adaptação com stakeholders, inclusive do que já está em produção.	**Portão de liberação** Não existe: liberar é decisão de valor do Product Owner para Incremento que atende à Definição de Pronto.
Sprint Review Lugar dos stakeholders: colaborar sobre Incremento e próximos passos.	**Daily Scrum e Sprint Retrospective** Sem stakeholders: Daily é dos Developers, Retrospective é do Scrum Team.
Variar escopo Resposta certa para data fixa, prazo apertado ou pedido de mais escopo.	**Variar qualidade ou duração da Sprint** Nunca: Definição de Pronto e duração fixa não entram no corte.
Velocidade usada pelo time Apoio para o próprio time prever.	**Velocidade comparada entre times** Inadequado: cada time estima de forma própria.

Números que caem

Item	Valor
Exemplo de previsão: histórico de 8 a 12 itens por Sprint e cerca de 40 itens restantes	**Entre quatro e cinco Sprints, revisando a cada Sprint**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** O Incremento só pode ser liberado depois de aprovado na Sprint Review.
Como não cair: Atendeu à Definição de Pronto e é utilizável, pode ser liberado a qualquer momento da Sprint; a Review não é portão.

⚠ **Alerta de prova:** Patrocinador pede data garantida: dar a data para preservar a confiança.
Como não cair: Apresente previsão baseada no histórico e atualize a cada Sprint; previsão sim, garantia não.

⚠ **Alerta de prova:** Data fixa por lei: aumentar horas, reduzir testes ou estender a Sprint.
Como não cair: Fixe a data e ajuste o escopo, priorizando o essencial; qualidade não varia.

⚠ **Alerta de prova:** Stakeholder deve participar da Daily Scrum para acompanhar o andamento, ou da Retrospective para sugerir melhorias.
Como não cair: Daily é dos Developers e Retrospective do Scrum Team; colaboração com stakeholders acontece na Sprint Review.

⚠ **Alerta de prova:** Comparar a velocidade de três times para escolher o melhor.
Como não cair: Cada time estima de forma própria; velocidade serve ao time para prever.

⚠ **Alerta de prova:** Previsões que mudam a cada Sprint indicam falha de planejamento.
Como não cair: Previsões melhoram com dados novos; isso é transparência.

⚠ **Alerta de prova:** Item quase pronto pode ser mostrado na Sprint Review para gerar feedback.
Como não cair: Só o que atende à Definição de Pronto é apresentado; o restante volta ao Product Backlog.

⚠ **Alerta de prova:** Comitê de mudanças ou janela trimestral: contornar a regra ou aceitar sem discutir.
Como não cair: Torne visível o impacto no valor e no aprendizado e trate com a organização.

⚠ **Alerta de prova:** Pedido de relatório semanal detalhado: criar o relatório ou reuniões extras.
Como não cair: Responda com mais transparência dos artefatos, não com mais reuniões ou relatórios individuais.

⚠ **Alerta de prova:** Developer falar direto com usuário é proibido, pois só o Product Owner fala com stakeholders.
Como não cair: O Scrum Team colabora com stakeholders, mantendo o Product Owner informado; decisões de valor continuam com ele.

⚠ **Alerta de prova:** Vários times no mesmo produto: cada um com sua Definição de Pronto e sua Review separada.
Como não cair: Mesma Definição de Pronto e Incremento integrado inspecionado com os stakeholders.

⚠ **Alerta de prova:** Stakeholder importante faltou: adiar ou cancelar a Sprint Review.
Como não cair: Eventos não são adiados por ausência; o Product Owner busca o feedback por outros meios.

⚠ **Alerta de prova:** Item ainda não refinado: dar uma data exata para tranquilizar o stakeholder.
Como não cair: Item não refinado não tem previsão confiável; refine antes de prever.

Como cai na prova

📝 **Como cai:** Stakeholder ou patrocinador pede data garantida, plano de 12 meses ou roadmap com datas fixas.
Como resolver: Escolha a resposta com visão, Meta do Produto e previsão baseada em dados, revista com frequência (roadmap por objetivos, horizontes mais incertos adiante). Elimine garantia, contrato de datas e plano item por item.

📝 **Como cai:** Cálculo simples de previsão a partir do histórico.
Como resolver: Divida o trabalho restante pelo intervalo do histórico (40 itens / 12 e 8 por Sprint), arredonde para uma faixa e escolha a alternativa que inclui revisão a cada Sprint.

📝 **Como cai:** Incremento pronto e pergunta se pode ser liberado ou mostrado.
Como resolver: Verifique a Definição de Pronto. Atende: pode ser liberado a qualquer momento e a Review pode inspecioná-lo. Não atende: não é liberado nem apresentado e volta ao Product Backlog.

📝 **Como cai:** Organização impõe barreira à entrega (comitê, janela, liberação semestral).
Como resolver: Marque a alternativa que torna visível o custo do atraso no valor e no aprendizado e busca reduzi-lo com a organização.

📝 **Como cai:** Escolha DUAS sobre previsão ou colaboração com stakeholders.
Como resolver: Prefira dados reais, revisão frequente, transparência do Product Backlog, Incremento inspecionado na Review e explicação de trade-offs. Descarte promessas, aceitar tudo, horas e status individuais.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
Incremento pronto no meio da Sprint, problema urgente de clientes	**Pode ser entregue antes do fim da Sprint.**

Escopo total aumenta a cada Sprint, mostrar aos stakeholders	**Burn-up com linha de escopo.**
Onde o trabalho está se acumulando	**Diagrama de fluxo cumulativo.**
Práticas citadas no Scrum Guide para prever progresso	**Burn-downs, burn-ups e fluxos cumulativos, sem substituir o empirismo.**
Quando a Meta do Produto será alcançada? Data para um conjunto de itens	**Previsão em faixa, baseada no progresso real, revisada a cada Sprint.**
Lei ou campanha com data fixa	**Data fixa, escopo variável, qualidade mantida.**
Exigências regulatórias em todo Incremento	**Critérios regulatórios na Definição de Pronto.**
Não expor a funcionalidade a todos ainda	**Liberar desligada e ativar aos poucos por grupos.**
Cliente quer ver o produto a cada duas semanas com Sprints de um mês	**Avaliar Sprints mais curtas ou entregas antes do fim da Sprint.**
Stakeholders só veem o produto a cada três meses ou só na versão final	**Feedback tarde demais: participar da Sprint Review a cada Sprint.**
Stakeholder influente e pouco presente	**Manter informado sobre decisões relevantes.**
Novo Product Owner com dezenas de stakeholders	**Identificar stakeholders, interesses, influência e forma de colaborar.**
Especialista em segurança para orientar a Sprint Planning	**Permitido: o Scrum Team pode convidar pessoas para orientar.**

🎯 **Se você lembrar de só uma coisa deste capítulo...**

- A Sprint Review não é portão de liberação: Incremento pronto pode ir para os usuários a qualquer momento.
- Previsão é baseada em dados, em faixa e revista a cada Sprint; nunca garantia.
- Data fixa: escopo varia, qualidade e duração da Sprint não.
- Stakeholders colaboram na Sprint Review, não na Daily Scrum nem na Retrospective.
- Velocidade não serve para comparar times.

Essenciais para a prova (checklist)

- ☐ Sei quem participa da Sprint Review e o que é inspecionado e adaptado nela.
- ☐ Sei o que burn-down, burn-up e fluxo cumulativo mostram e por que não substituem o empirismo.
- ☐ Sei fazer uma previsão em faixa a partir do histórico do time.
- ☐ Sei responder a pedidos de data garantida, relatório extra e plano anual detalhado.

- ☐ Sei quando um Incremento pode ser liberado e quem decide liberar.
- ☐ Sei lidar com data fixa, mais escopo sem mudar data e preço e prazo fixos.
- ☐ Sei como vários Scrum Teams entregam e revisam um Incremento integrado com a mesma Definição de Pronto.
- ☐ Sei por que comparar velocidade entre times é inadequado.
- ☐ Sei como tratar restrições organizacionais que atrasam valor.

Questões de revisão do capítulo

R4.1 · Stakeholders, Previsão e Entregas

O Product Owner usa um gráfico de burn-up para conversar com stakeholders sobre uma versão. O que esse gráfico mostra?

A. A lista de impedimentos resolvidos pelo Scrum Master na Sprint
B. A comparação de produtividade entre os Scrum Teams da empresa ao longo do ano
C. O trabalho concluído ao longo do tempo em relação ao escopo total previsto
D. As horas trabalhadas por cada Developer em cada dia da Sprint atual

R4.2 · Stakeholders, Previsão e Entregas

Durante uma versão, o escopo total aumenta a cada Sprint. Qual gráfico deixa isso mais visível para os stakeholders?

A. Burn-down sem linha de escopo
B. Burn-up com linha de escopo total
C. Lista de impedimentos da Sprint
D. Relatório de horas por pessoa

R4.3 · Stakeholders, Previsão e Entregas

Os stakeholders reclamam que as previsões mudam a cada Sprint. Como o Product Owner deve explicar?

A. Previsões não fazem parte do Scrum e serão abandonadas
B. As previsões mudam porque os Developers não se esforçam o suficiente
C. A partir de agora, a previsão não mudará mais até o fim da versão
D. Previsões melhoram com dados novos, o que aumenta a transparência

R4.4 · Stakeholders, Previsão e Entregas

O Product Owner percebe que a Meta do Produto não será alcançada na data que os stakeholders esperam. Qual atitude é mais adequada?

A. Reduzir a Definição de Pronto para acelerar as entregas
B. Informar os stakeholders cedo, mostrando os dados e as opções disponíveis
C. Esperar a data chegar para evitar desgaste antecipado com os stakeholders
D. Pedir horas extras aos Developers até recuperar o atraso

Respostas comentadas

R4.1. C) O trabalho concluído ao longo do tempo em relação ao escopo total previsto

Por quê: Burn-up é uma das práticas de previsão citadas no Scrum Guide. Ele mostra o trabalho concluído ao longo do tempo e o escopo total, o que ajuda a prever quando uma meta pode ser alcançada e a discutir trade-offs.

✗ A) Incorreta. O burn-up não é uma lista de impedimentos.
✗ B) Incorreta. Burn-ups não servem para comparar produtividade entre times, que trabalham em contextos diferentes.
✗ D) Incorreta. O burn-up não acompanha horas individuais.

> **Dica de prova:** Burn-up mostra progresso e mudanças de escopo no mesmo gráfico. É útil em conversas com stakeholders.

R4.2. B) Burn-up com linha de escopo total

Por quê: Burn-ups são uma das práticas de previsão citadas no Scrum Guide. Com a linha de escopo total, mostram se o atraso vem de ritmo ou de aumento de escopo.

✗ A) Incorreta. Um burn-down sem linha de escopo mistura progresso e mudança de escopo, o que esconde o aumento.
✗ C) Incorreta. Impedimentos não mostram a evolução do escopo.
✗ D) Incorreta. Horas por pessoa não mostram escopo nem progresso.

> **Dica de prova:** Escopo mudando? Burn-up com linha de escopo mostra isso com clareza.

R4.3. D) Previsões melhoram com dados novos, o que aumenta a transparência

Por quê: Práticas de previsão são úteis, mas não substituem o empirismo. Em ambientes complexos, o que vai acontecer é desconhecido, e só o que já aconteceu pode ser usado para decidir.

✗ A) Incorreta. O Scrum Guide cita práticas de previsão como úteis.
✗ B) Incorreta. Culpar os Developers não explica a natureza das previsões e prejudica a confiança.
✗ C) Incorreta. Congelar a previsão esconde a realidade e reduz a transparência.

> **Dica de prova:** Previsão que muda com dados é sinal de transparência, não de falha.

R4.4. B) Informar os stakeholders cedo, mostrando os dados e as opções disponíveis

Por quê: Transparência é pilar do Scrum. Ao perceber um desvio, o Product Owner informa os stakeholders cedo e usa dados para discutir opções.

✗ A) Incorreta. A qualidade não diminui. A Definição de Pronto não é reduzida para acelerar entregas.

✗ C) Incorreta. Esconder a informação reduz a transparência e costuma gerar um problema maior depois.

✗ D) Incorreta. Horas extras constantes prejudicam a qualidade e não são uma resposta sustentável.

> **Dica de prova:** Más notícias cedo e com dados: é assim que o Product Owner preserva a confiança.

05

CAPÍTULO 5

Valor, Visão e Estratégia de Produto

Aqui está o coração do PSPO I: maximizar valor é fazer as escolhas certas, não fazer mais. Preciso dominar output x outcome, hipóteses e experimentos, visão, Meta do Produto e estratégia, e o Evidence-Based Management (EBM) com as quatro áreas-chave de valor e os três níveis de metas. A prova traz muitos cenários de medida para classificar na área-chave certa e situações em que dados contradizem o plano. Onde mais erro: colocar medidas de tempo de reparo ou tempo para aprender em Capacidade de Inovar, e não Tempo para o Mercado, e aceitar alternativas que medem valor por funcionalidades, horas, velocidade ou prazo cumprido.

Neste capítulo
18 tópicos-chave · 13 pegadinhas · 7 macetes · 14 gatilhos · 4 questões comentadas

Tópicos-chave

Maximizar valor

Obter o maior resultado possível com os recursos disponíveis: escolher bem, não produzir mais. Quem confirma o valor, em última instância, são clientes e usuários pelo uso e pelos resultados. O Scrum Guide não define como medir valor, porque isso varia entre organizações, times e pessoas; o EBM complementa.

Output x outcome

Output é o que foi entregue (funcionalidades, linhas de código, itens); outcome é a mudança de resultado gerada para cliente ou negócio, como tempo médio de entrega reduzido em 15%. Mais output não garante mais outcome. Entregar no prazo e sem valor ainda é fracasso de produto.

Evidence-Based Management (EBM)

Guia da Scrum org para medir valor e decidir com evidências, aplicando o empirismo: definir meta, formular hipótese, experimentar, medir e adaptar. Recomenda observar as quatro áreas-chave de valor em conjunto, para equilibrar valor atual e futuro. Não serve para comparar produtividade entre times nem mede papéis do Scrum.

Valor Atual (Current Value)

Valor que o produto entrega hoje. Medidas: satisfação dos clientes, satisfação dos funcionários (pessoas satisfeitas tendem a entregar melhor valor), receita por funcionário e índice de uso pelos clientes (Customer Usage Index).

Valor Não Realizado (Unrealized Value)

Potencial de valor ainda não capturado. Indicadores: participação de mercado, grande demanda não atendida, lacuna entre a experiência que os clientes desejam e a que o produto oferece. Ajuda a decidir investimentos futuros e a conquista de novos clientes.

Tempo para o Mercado (Time to Market)

Rapidez para entregar novas capacidades e aprender com elas. Medidas: frequência de releases, lead time, tempo de ciclo (cycle time), tempo médio para restaurar o serviço (mean time to repair) e tempo para aprender (Time to Learn). Concorrência lançando mais rápido é sinal de atenção aqui.

Capacidade de Inovar (Ability to Innovate)

Eficácia para entregar novas capacidades. Medidas: dívida técnica, tendência de defeitos, taxa de inovação (Innovation Rate, parcela do esforço em novas capacidades e não em manutenção), índice de tempo dedicado ao produto (On-Product Index), tempo gasto com trocas de contexto, tempo com manutenção e defeitos e muitas versões antigas em uso. Esforço alto em correções reduz esta área.

Metas no EBM

Meta Estratégica: objetivo ambicioso de longo prazo rumo à visão. Metas Intermediárias: marcos que indicam progresso rumo à estratégica; a Meta do Produto pode funcionar como uma delas. Meta Tática Imediata: o passo de agora, como a Meta da Sprint.

Hipótese de valor

Torna explícita a suposição a ser testada; não garante sucesso nem dispensa medição. Boa hipótese é mensurável: resultado esperado, quanto e até quando (ex.: reduzir o abandono do carrinho em 10% em dois meses). Até a ideia do Product Owner, sem evidência, é hipótese.

Experimento pequeno

Com incerteza alta ou custo alto e valor incerto, entregue a menor versão que permita medir o interesse real, meça e adapte. Para escolher entre opções, libere versões para grupos diferentes e compare. Defina antes o resultado esperado e como será medido; especificar tudo e entregar de uma vez é a alternativa errada.

Visão do produto

Propósito: para onde e por quê, clara e inspiradora, explicando o problema e para quem o produto existe. É estável e não é documento de requisitos. Sem visão, o Product Backlog vira lista de pedidos; um novo Product Owner começa construindo e comunicando a visão com stakeholders e o time.

Meta do Produto útil

Descreve um resultado desejado e permite verificar o progresso, como reduzir pela metade o tempo para abrir uma conta. Lista de funcionalidades, aumento de velocidade ou migração técnica não são boas metas; metas vagas como ser o número 1 do mercado não orientam decisões. Verifica-se com evidências de resultado, não contagem de itens.

Estratégia de produto

Define como o produto vai alcançar a visão e gerar valor para o negócio, ligando a Meta do Produto aos objetivos da organização. Deve ser revista com base em evidências: se após várias Sprints os resultados não vêm, adapta-se estratégia e Meta do Produto, não se aumenta a produção.

Produto

Segundo o Scrum Guide 2020, veículo para entregar valor, com fronteira clara, stakeholders conhecidos e usuários bem definidos. Pode ser serviço, produto físico ou algo abstrato; não é sinônimo de software nem de projeto. Produto interno também tem usuários, cujo valor deve ser maximizado.

Cliente, usuário e stakeholder

Cliente compra, usuário usa, stakeholder tem interesse; podem ser pessoas diferentes, como a empresa que compra a folha de pagamento e as pessoas que a operam. O Product Owner equilibra os interesses para maximizar o valor total. Em órgão público, valor pode ser benefício ao cidadão, redução de custos ou impacto social.

Custo do atraso e custo passado

Custo do atraso alto (multa após data legal) antecipa o item; conformidade também é valor. Custo passado não justifica investimento futuro: abandonar uma Meta do Produto ou funcionalidade pode ser a melhor decisão. Valor também se cria removendo ou simplificando o que é pouco usado e caro de manter.

Métrica de vaidade

Métrica que cresce, mas não muda decisões nem reflete valor, como downloads com retenção de 2%. Boa medida muda decisões sobre o produto; prefira uso, retenção e satisfação. Uma métrica só (só satisfação atual) ignora ameaças futuras.

Incremento utilizável e liberação

Para gerar valor, o Incremento precisa ser utilizável, e o valor só se realiza quando os usuários o usam. Pronto e parado há meses não gera valor nem feedback. Liberar com frequência traz valor antes e feedback para as próximas decisões.

Macetes para memorizar

💡 Áreas-chave do EBM

Hoje (Atual), Potencial (Não Realizado), Rapidez (Tempo para o Mercado), Freio (Capacidade de Inovar).

💡 Tempo para o Mercado x Capacidade de Inovar

Mede a rapidez de entregar ou aprender (releases, lead time, cycle time, tempo de reparo, tempo para aprender)? Tempo para o Mercado. Mede o que desvia esforço do novo (dívida técnica, defeitos, manutenção, trocas de contexto, tempo fora do produto)? Capacidade de Inovar, mesmo que a medida seja em tempo.

💡 Hierarquia de metas

Visão → Meta Estratégica → Meta do Produto (intermediária) → Meta da Sprint (tática imediata): do longo ao curto prazo.

💡 Output x outcome

Output sai da fábrica; outcome muda a vida do cliente.

💡 Ciclo do EBM

Meta, hipótese, experimento, medição, adaptação: empirismo aplicado ao valor.

💡 Decisão com dados

Dados contradizem o plano? Adapte. A alternativa que usa evidências costuma ser a certa.

💡 Custo passado

O dinheiro gasto não volta; decida pelo valor futuro.

Tabela de memorização: não confunda

Quando é...	...e quando é
Valor Atual O que o produto entrega hoje: satisfação, uso, receita por funcionário; foco em reter clientes atuais.	**Valor Não Realizado** Potencial ainda não capturado: participação de mercado, demanda não atendida, lacuna de experiência; foco em novos clientes.
Tempo para o Mercado Rapidez para entregar e aprender: frequência de releases, lead time, cycle time, tempo de reparo, tempo para aprender.	**Capacidade de Inovar** O que freia a inovação: dívida técnica, defeitos, taxa de inovação, On-Product Index, tempo gasto com trocas de contexto ou com manutenção, versões antigas.
Output O que o time produziu: funcionalidades, itens, entregas no prazo.	**Outcome** Mudança de resultado para cliente ou negócio: menos chamadas ao suporte, entrega mais rápida, mais retenção.

Visão Propósito amplo e duradouro, estável e inspirador.	**Meta do Produto** Objetivo concreto e mensurável que aproxima o produto da visão; cumprido ou abandonado.
Meta do Produto Longo prazo; pode funcionar como Meta Intermediária do EBM.	**Meta da Sprint** Passo de uma Sprint rumo à Meta do Produto; exemplo de Meta Tática Imediata.
Meta Estratégica Objetivo ambicioso de longo prazo que leva à visão.	**Metas Intermediárias** Marcos que indicam progresso rumo à Meta Estratégica.

Números que caem

Item	Valor
Áreas-chave de valor do EBM	**4: Valor Atual, Valor Não Realizado, Tempo para o Mercado, Capacidade de Inovar**
Níveis de metas do EBM	**3: Estratégica, Intermediárias, Táticas Imediatas**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Tempo médio para restaurar o serviço é Capacidade de Inovar, porque fala de falhas.
Como não cair: Qualquer medida de tempo de resposta, entrega ou aprendizado fica em Tempo para o Mercado, inclusive mean time to repair e Time to Learn.

⚠ **Alerta de prova:** Satisfação dos funcionários é Capacidade de Inovar ou não pertence ao EBM.
Como não cair: Satisfação dos funcionários é medida de Valor Atual, complementando (sem substituir) as medidas de clientes.

⚠ **Alerta de prova:** Trocas de contexto ou On-Product Index ficam em Tempo para o Mercado, porque atrasam as entregas.
Como não cair: Tudo o que tira o time do produto é medido em Capacidade de Inovar.

⚠ **Alerta de prova:** Velocidade do time, qualidade do código ou satisfação do Scrum Master como área-chave do EBM.
Como não cair: Só quatro áreas: Valor Atual, Valor Não Realizado, Tempo para o Mercado e Capacidade de Inovar. Velocidade nem é medida do EBM.

⚠ **Alerta de prova:** Taxa de inovação é o número de ideias sugeridas pelos stakeholders.
Como não cair: É a parcela do esforço dedicada a novas capacidades, e não à manutenção.

⚠ **Alerta de prova:** O problema da Meta do Produto 'ser o aplicativo número 1 do mercado' é ser ambiciosa demais.
Como não cair: Metas podem ser ambiciosas; o problema é ser vaga e difícil de medir, o que dificulta orientar decisões.

⚠ **Alerta de prova:** Sem receita, como em órgão público, o Product Owner maximiza a produção de funcionalidades.
Como não cair: Valor pode ser benefício ao cidadão, redução de custos ou impacto social. Reduzir valor a receita ou quantidade de funcionalidades é sinal de alternativa errada.

⚠ **Alerta de prova:** Recurso pouco usado: manter o plano aprovado, adicionar funcionalidades ou esperar o fim do ano.
Como não cair: Use a evidência para reavaliar o recurso e reordenar o Product Backlog o quanto antes.

⚠ **Alerta de prova:** Item que atende à Definição de Pronto já gera todo o valor previsto, ou valor contábil.
Como não cair: O valor só se realiza quando os usuários podem usar o Incremento.

⚠ **Alerta de prova:** Item de custo maior deve vir antes, pois custo alto indica mais valor.
Como não cair: Mesmo valor mais cedo e mais barato vence; custo alto não indica valor alto.

⚠ **Alerta de prova:** Horas já gastas ou opinião de quem implementou justificam continuar investindo em uma funcionalidade.
Como não cair: Custo passado não decide o futuro; olhe dados de uso e resultado.

⚠ **Alerta de prova:** A visão deve ser detalhada como documento de requisitos e fixar o escopo, ou mudar a cada Sprint.
Como não cair: Visão é estável e inspiradora, não especificação; o que se adapta com frequência é o Product Backlog.

⚠ **Alerta de prova:** A Meta do Produto é a soma das Metas da Sprint, ou a visão a substitui em produtos pequenos.
Como não cair: A Meta do Produto orienta as Sprints e é obrigatória, independentemente do porte.

Como cai na prova

📝 **Como cai:** Uma medida do EBM é citada (às vezes pelo nome em inglês) e pede-se a área-chave.
Como resolver: Pergunte: é sobre hoje (Atual), potencial de mercado (Não Realizado), rapidez para entregar e aprender (Tempo para o Mercado) ou o que freia o time, como dívida técnica, defeitos e tempo fora do produto (Capacidade de Inovar)? Distratores trazem as outras três áreas.

📝 **Como cai:** Escolher a melhor Meta do Produto ou Meta da Sprint entre várias.
Como resolver: Marque a que descreve resultado para usuário ou negócio e é verificável. Elimine listas de funcionalidades, velocidade, migração técnica e metas vagas.

📝 **Como cai:** Dados mostram que algo não funciona como o Product Owner esperava.
Como resolver: Escolha a alternativa que usa a evidência para adaptar (reordenar, reavaliar, remover, rever estratégia). Elimine manter o plano, produzir mais ou esperar o fim do período.

📝 **Como cai:** Decidir onde investir ou como validar uma ideia.
Como resolver: Prefira evidências de necessidade real, experimento pequeno com medição definida antes e aprendizado rápido. Elimine velocidade do time, quantidade de Developers disponíveis, cópia de funcionalidades do concorrente e entrega única grande.

📝 **Como cai:** Gestor mede sucesso por funcionalidades, prazo, horas ou velocidade.
Como resolver: Aponte a limitação: mede output ou esforço, não resultado e valor.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
Satisfação dos clientes, uso real hoje, receita por funcionário	**Valor Atual.**
Participação de mercado, demanda não atendida, lacuna entre desejado e entregue	**Valor Não Realizado.**
Frequência de releases, lead time, cycle time, tempo de reparo, tempo para validar ideia, concorrência mais rápida	**Tempo para o Mercado.**
Dívida técnica, defeitos, 70% do esforço em correções, versões antigas, alternar entre projetos	**Capacidade de Inovar.**
Produto maduro com boa receita perdendo clientes para inovadores	**Valor Não Realizado e Capacidade de Inovar.**
Novos clientes ou reter os atuais	**Valor Não Realizado e Valor Atual.**
Objetivo ambicioso de longo prazo rumo à visão	**Meta Estratégica (EBM).**
Exemplo de Meta Tática Imediata	**A Meta da Sprint.**

Incerteza alta, custo alto, valor incerto	**Versão mínima, medir e adaptar.**
Duas versões de uma tela, qual converte mais	**Liberar para grupos diferentes e comparar.**
1 milhão de downloads, 2% de retenção	**Métrica de vaidade; investigar a baixa retenção.**
Valor parecido entre duas iniciativas	**Escolha a que permite aprender mais rápido.**
Startup com pouca verba e muitas hipóteses	**Ordenar primeiro experimentos das hipóteses mais arriscadas.**
Diretor avalia por funcionalidades entregues por Sprint	**Mede output, não outcome nem valor.**

🎯 **Se você lembrar de só uma coisa deste capítulo...**

- Tempo de reparo e tempo para aprender são Tempo para o Mercado, não Capacidade de Inovar.
- Satisfação dos funcionários é Valor Atual.
- Meta da Sprint é exemplo de Meta Tática Imediata; Meta do Produto pode ser Meta Intermediária.
- Valor é outcome verificado por evidências de uso, nunca contagem de funcionalidades, horas ou velocidade.
- Custo passado não justifica continuar: abandonar meta ou remover funcionalidade pode maximizar valor.

Essenciais para a prova (checklist)

- ☐ Sei classificar qualquer medida citada nas quatro áreas-chave de valor do EBM.
- ☐ Sei explicar a diferença entre output e outcome com exemplo.
- ☐ Sei os três níveis de metas do EBM e onde entram Meta do Produto e Meta da Sprint.
- ☐ Sei escrever uma hipótese mensurável e o ciclo meta, hipótese, experimento, medição, adaptação.
- ☐ Sei distinguir visão, estratégia, Meta do Produto e Meta da Sprint.
- ☐ Sei reconhecer uma boa Meta do Produto e por que listas de funcionalidades e metas vagas falham.
- ☐ Sei a definição de produto do Scrum Guide 2020 e a diferença entre cliente, usuário e stakeholder.
- ☐ Sei por que um Incremento pronto e não liberado não gera valor.
- ☐ Sei identificar métrica de vaidade e decisões contaminadas por custo passado.

Questões de revisão do capítulo

R5.1 · Valor, Visão e Estratégia de Produto

Qual é o risco de um Product Owner definir a Meta do Produto como uma lista de funcionalidades?

A. Impedir que os Developers estimem os itens do backlog
B. Focar em produção e perder de vista o resultado
C. Obrigar o time a usar histórias de usuário
D. Tornar a meta mensurável demais para os stakeholders

R5.2 · Valor, Visão e Estratégia de Produto

Qual é o papel das hipóteses na gestão de valor feita pelo Product Owner?

A. Substituir a necessidade de medir resultados
B. Tornar explícitas as suposições que serão testadas
C. Garantir antecipadamente que as funcionalidades terão sucesso no mercado
D. Definir as tarefas técnicas da Sprint

R5.3 · Valor, Visão e Estratégia de Produto

Por que o EBM inclui a satisfação dos funcionários entre as medidas de Valor Atual?

A. Porque define o salário de cada Developer
B. Porque substitui a necessidade de medir a satisfação dos clientes e o uso do produto
C. Porque é a única medida exigida pelo Scrum Guide
D. Porque pessoas satisfeitas tendem a entregar melhor valor aos clientes

R5.4 · Valor, Visão e Estratégia de Produto

Um Scrum Team desenvolve uma plataforma usada somente por outros times da própria empresa. Cujo valor o Product Owner deve maximizar?

A. Dos times internos que usam a plataforma para entregar valor
B. De ninguém, pois o produto não é vendido a clientes
C. Dos Developers do próprio Scrum Team que constroem a plataforma
D. Da diretoria que aprovou o investimento na plataforma

Respostas comentadas

R5.1. B) Focar em produção e perder de vista o resultado

Por quê: A Meta do Produto descreve um estado futuro do produto que serve de alvo para o planejamento. Metas orientadas a resultado ajudam a decidir o que construir e o que deixar de fora.

✗ A) Incorreta. A forma da Meta do Produto não impede os Developers de estimar.
✗ C) Incorreta. A Meta do Produto não obriga nenhum formato de item.
✗ D) Incorreta. Ser mensurável é qualidade de uma boa meta, não um risco.

> **Dica de prova:** Meta do Produto boa descreve resultado; lista de funcionalidades é backlog, não meta.

R5.2. B) Tornar explícitas as suposições que serão testadas

Por quê: O EBM trata metas e entregas como hipóteses a serem testadas com experimentos e medições. Isso aplica o empirismo à gestão de valor.

✗ A) Incorreta. Hipóteses exigem medição, não a substituem.
✗ C) Incorreta. Hipóteses admitem a incerteza. Não garantem sucesso.
✗ D) Incorreta. Tarefas técnicas são planejadas pelos Developers.

> **Dica de prova:** Hipótese explícita + medição = decisão baseada em evidência.

R5.3. D) Porque pessoas satisfeitas tendem a entregar melhor valor aos clientes

Por quê: Valor Atual inclui medidas como satisfação de clientes, satisfação de funcionários, receita por funcionário e uso pelos clientes. Pessoas engajadas tendem a entregar mais valor.

✗ A) Incorreta. O EBM não trata de remuneração.
✗ B) Incorreta. A satisfação dos funcionários complementa, mas não substitui, medidas de clientes e uso.
✗ C) Incorreta. O Scrum Guide não exige medidas do EBM.

> **Dica de prova:** Valor Atual olha clientes e também as pessoas que constroem o produto.

R5.4. A) Dos times internos que usam a plataforma para entregar valor

Por quê: O Scrum Guide define produto como um veículo para entregar valor, com fronteira clara, stakeholders conhecidos e usuários ou clientes bem definidos. Plataformas internas se encaixam nessa definição.

✗ B) Incorreta. Um produto não precisa ser vendido para ter usuários e gerar valor.
✗ C) Incorreta. Os Developers constroem o produto, mas não são os usuários da plataforma.
✗ D) Incorreta. A diretoria é stakeholder, mas o valor se realiza no uso pelos times internos.

> **Dica de prova:** Produto interno também tem usuários. Identifique-os e meça o valor que recebem.

DIA DA PROVA

Estratégia de prova

A prova PSPO I tem 80 questões em 60 minutos, com nota de corte de 85%, em múltipla escolha, múltipla resposta e verdadeiro/falso, aplicada em inglês. Ela cobra o Scrum Guide 2020 ao pé da letra e o pensamento de produto do Evidence-Based Management: valor, evidências e empirismo vencem planos fixos, produção e controle. Os enunciados são cenários curtos em que alguém (gerente, diretor, stakeholder, comitê) tenta tirar uma decisão do dono certo, e a resposta devolve a decisão a quem é responsável. Distratores típicos tornam obrigatória uma prática complementar, colocam o Scrum Master decidindo sobre o produto, estendem a Sprint, reduzem a qualidade ou prometem datas. Sempre que uma alternativa usa evidências, transparência e negociação de escopo, ela tende a ser a certa.

Como responder

✓ Ritmo: são 80 questões em 60 minutos, menos de um minuto por questão; responda as diretas (timebox, compromissos, áreas do EBM) em segundos para sobrar tempo aos cenários.

✓ Antes de ler as alternativas, identifique de quem é a decisão no cenário: Product Owner (valor, ordem, cancelamento, liberação), Developers (como, quem, quanto, Sprint Backlog) ou Scrum Team (Meta da Sprint, Definição de Pronto).

✓ Em múltipla resposta, conte quantas o enunciado pede (DUAS, TRÊS) e valide cada alternativa como verdadeiro ou falso isoladamente contra o Scrum Guide.

✓ Elimine na hora alternativas com Sprint Zero, Sprint estendida, subtimes, comitê de Product Owners, Scrum Master ordenando o backlog, regra de 10%, Definition of Ready ou três perguntas obrigatórias.

✓ Como a prova é em inglês, memorize os termos originais: Product Goal, Sprint Goal, Developers, accountability, Current Value, Unrealized Value, Time to Market, Ability to Innovate.

✓ Em cenário de problema durante a Sprint, pergunte primeiro se a Meta da Sprint continua válida: válida, renegocie escopo; obsoleta, o Product Owner pode cancelar.

✓ Em questões de valor, escolha a alternativa que mede outcome e usa evidências, e descarte as que medem funcionalidades, horas, velocidade ou prazo cumprido.

✓ Desconfie de palavras absolutas como sempre, somente, proibido e obrigatório quando aplicadas a práticas que o Scrum Guide não define.

Erros que derrubam candidatos

✗ Responder pela prática do dia a dia da empresa, e não pelo texto do Scrum Guide 2020.

✗ Confundir pilares (Transparência, Inspeção, Adaptação) com valores (Compromisso, Foco, Abertura, Respeito, Coragem).

✗ Achar que a Daily Scrum encolhe com Sprints curtas ou que o Scrum Master precisa conduzir todos os eventos.

✗ Classificar tempo de reparo ou tempo para aprender em Capacidade de Inovar em vez de Tempo para o Mercado.

✗ Tratar a Sprint Review como aprovação formal ou portão de liberação do Incremento.

✗ Acreditar que delegar trabalho transfere a responsabilidade do Product Owner, ou que vários times exigem vários Product Owners.

✗ Aceitar garantia de data ou redução de qualidade para cumprir prazo, em vez de negociar escopo.

Na semana da prova

- ☐ Releia o Scrum Guide 2020 inteiro pelo menos uma vez, marcando as palavras exatas de cada evento, artefato, compromisso e responsabilidade.
- ☐ Revise a tabela das quatro áreas-chave de valor do EBM com as medidas de cada uma e os três níveis de metas.
- ☐ Faça as provas completas cronometradas em 60 minutos e só avance quando passar com folga acima de 85%.
- ☐ Revise cada erro identificando qual distrator enganou você (dono errado da decisão, prática complementar, valor x pilar) e anote a regra.
- ☐ Treine a leitura dos termos em inglês para não perder tempo traduzindo na hora.
- ☐ Na véspera, revise só os macetes, os números (8-4-3-15, 10 pessoas ou menos, 3 compromissos) e as listas de lembrar.

REVISÃO FINAL

Folha de cola

Leia na véspera e na manhã da prova: os gatilhos e as pegadinhas mais importantes de cada domínio.

1. Gestão do Product Backlog

‣ Quando o refinamento deve acontecer? Quem decide como refinar? → **Atividade contínua durante a Sprint; técnicas, frequência e duração decididas pelo Scrum Team.**

‣ Quem dimensiona / estima os itens? → **Os Developers que farão o trabalho.**

‣ Item no topo há várias Sprints e nunca é selecionado → **Precisa ser refinado ou ter o valor reavaliado.**

‣ Valor parecido, mas um item reduz grande risco técnico → **Antecipar o item de risco: risco e aprendizado desempatam.**

‣ Item de alto valor depende de outro de baixo valor → **O de baixo valor sobe antes, porque habilita o valioso.**

⚠ Refinamento é contínuo, ao longo da Sprint, conforme a necessidade. Toda Sprint gera Incremento utilizável e o Scrum Guide não define reunião fixa de refinamento.

⚠ A regra de 10% era de versões antigas e não aparece no Scrum Guide 2020. Refinamento não é evento e não tem timebox.

⚠ Quem faz, estima: os Developers dimensionam. O Product Owner ordena e explica; o Scrum Master pode facilitar, mas não estima.

2. O Framework Scrum

‣ Sprint de um mês: timebox de Planning, Review ou Retrospective → **8, 4 e 3 horas; Daily Scrum sempre 15 minutos.**

‣ Decisões baseadas no que é observado, problema complexo e incerto → **Empirismo (e pensamento enxuto: menos desperdício, foco no essencial).**

‣ Informação escondida ou maquiada → **Pilar Transparência; valor Abertura.**

‣ Alertar risco sério sob pressão do patrocinador → **Coragem.**

‣ Ouvir o júnior com a mesma seriedade → **Respeito.**

⚠ Pilares são só Transparência, Inspeção e Adaptação. Respeito é valor; colaboração e previsão não são pilares.

⚠ Daily Scrum tem sempre 15 minutos. Só Planning, Review e Retrospective costumam ter timebox menor em Sprints mais curtas.

⚠ Os cinco eventos são Sprint, Sprint Planning, Daily Scrum, Sprint Review e Sprint Retrospective. Refinamento é atividade; Sprint Zero não existe.

3. Responsabilidades do Product Owner

‣ Principal responsabilidade (accountability) do Product Owner → **Maximizar o valor do produto resultante do trabalho do Scrum Team.**

‣ Analista de negócios escreve itens a pedido do Product Owner → **Permitido; responsabilidade pelo Product Backlog continua com o Product Owner.**

‣ Quem decide liberar um Incremento pronto → **O Product Owner (decisão de valor).**

‣ Quem pode cancelar a Sprint → **Somente o Product Owner, com Meta da Sprint obsoleta.**

‣ Onde ficam visíveis as decisões do Product Owner → **Conteúdo e ordem do Product Backlog e Incremento inspecionável na Sprint Review.**

⚠ Delegou trabalho, não responsabilidade: o Product Owner continua respondendo pelo valor.

⚠ O Product Owner é uma pessoa; comitê pode representar necessidades, não substituí-lo.

⚠ Escalar não multiplica Product Owners: um produto, um Product Owner, que pode delegar trabalho.

4. Stakeholders, Previsão e Entregas

‣ Incremento pronto no meio da Sprint, problema urgente de clientes → **Pode ser entregue antes do fim da Sprint.**

‣ Escopo total aumenta a cada Sprint, mostrar aos stakeholders → **Burn-up com linha de escopo.**

‣ Onde o trabalho está se acumulando → **Diagrama de fluxo cumulativo.**

‣ Práticas citadas no Scrum Guide para prever progresso → **Burn-downs, burn-ups e fluxos cumulativos, sem substituir o empirismo.**

‣ Quando a Meta do Produto será alcançada? Data para um conjunto de itens → **Previsão em faixa, baseada no progresso real, revisada a cada Sprint.**

⚠ Atendeu à Definição de Pronto e é utilizável, pode ser liberado a qualquer momento da Sprint; a Review não é portão.

⚠ Apresente previsão baseada no histórico e atualize a cada Sprint; previsão sim, garantia não.

⚠ Fixe a data e ajuste o escopo, priorizando o essencial; qualidade não varia.

5. Valor, Visão e Estratégia de Produto

‣ Satisfação dos clientes, uso real hoje, receita por funcionário → **Valor Atual.**

‣ Participação de mercado, demanda não atendida, lacuna entre desejado e entregue → **Valor Não Realizado.**

‣ Frequência de releases, lead time, cycle time, tempo de reparo, tempo para validar ideia, concorrência mais rápida → **Tempo para o Mercado.**

‣ Dívida técnica, defeitos, 70% do esforço em correções, versões antigas, alternar entre projetos → **Capacidade de Inovar.**

‣ Produto maduro com boa receita perdendo clientes para inovadores → **Valor Não Realizado e Capacidade de Inovar.**

⚠ Qualquer medida de tempo de resposta, entrega ou aprendizado fica em Tempo para o Mercado, inclusive mean time to repair e Time to Learn.

⚠ Satisfação dos funcionários é medida de Valor Atual, complementando (sem substituir) as medidas de clientes.

⚠ Tudo o que tira o time do produto é medido em Capacidade de Inovar.

GABARITO

Respostas do diagnóstico

D1. D) Maximizar o valor entregue e o aprendizado ao longo das Sprints → revise o capítulo 1

Por quê: Ordenar o Product Backlog é uma das atividades de gestão do Product Owner, que busca maximizar o valor do produto resultante do trabalho do Scrum Team.

✗ A) Incorreta. A ordem de chegada não reflete valor.

✗ B) Incorreta. A ordem não é definida pelas especialidades dos Developers.

✗ C) Incorreta. Distribuir igualmente entre áreas ignora onde está o maior valor.

> **Dica de prova:** A ordem do Product Backlog responde a uma pergunta: o que gera mais valor e aprendizado agora?

D2. B) Duas fontes de prioridade confundem o time e reduzem a transparência → revise o capítulo 1

Por quê: O Product Backlog é a única fonte do trabalho realizado pelo Scrum Team. Documentos complementares, como o roadmap, devem ser coerentes com ele.

✗ A) Incorreta. A ordem do trabalho está no Product Backlog. Um roadmap pode dar direção, mas não deve conflitar com ele.

✗ C) Incorreta. Os Developers não devem escolher entre fontes de prioridade. O Product Owner mantém uma única ordem.

✗ D) Incorreta. O Scrum Master não concilia prioridades de produto.

> **Dica de prova:** Uma única fonte de trabalho: o Product Backlog. Roadmap é previsão, não ordem paralela.

D3. C) Criar um item menor de investigação para reduzir a incerteza → revise o capítulo 1

Por quê: O Scrum Team é responsável também por experimentação e pesquisa. Itens de investigação reduzem a incerteza e tornam o refinamento mais preciso.

✗ A) Incorreta. O item continua no Product Backlog. A incerteza é tratada com aprendizado, não com exclusão.

✗ B) Incorreta. O dimensionamento é dos Developers que farão o trabalho.

✗ D) Incorreta. Um número sem base reduz a transparência e pode comprometer a Sprint.

> **Dica de prova:** Incerteza alta: primeiro aprenda com um item pequeno, depois estime.

D4. D) Aos principais stakeholders → revise o capítulo 2

Por quê: Na Sprint Review, o Scrum Team apresenta os resultados aos principais stakeholders e o progresso rumo à Meta do Produto é discutido.

✗ A) Incorreta. A diretoria pode ser stakeholder, mas o Scrum Guide fala em principais stakeholders, não em uma área específica.

✗ B) Incorreta. A Sprint Review não é avaliação de desempenho individual para gestores.

✗ C) Incorreta. A Sprint Review não é uma auditoria.

> **Dica de prova:** Sprint Review: resultados para os principais stakeholders e conversa sobre o próximo passo.

D5. D) O quanto antes, para minimizar desvios adicionais → revise o capítulo 2

Por quê: Adaptação: se algum aspecto de um processo sair dos limites aceitáveis ou se o produto resultante for inaceitável, o processo ou o que está sendo produzido deve ser ajustado o quanto antes para minimizar desvios adicionais.

✗ A) Incorreta. A Sprint Retrospective é uma oportunidade de adaptação, mas não a única. Desvios devem ser ajustados o quanto antes.

✗ B) Incorreta. Esperar o fim do trimestre aumenta o desvio e o desperdício.

✗ C) Incorreta. Adiar o ajuste para depois da release permite que o problema cresça.

> **Dica de prova:** Adaptação é imediata quando o desvio é detectado. Não espere o próximo evento se o problema já está claro.

D6. B) No empirismo e no pensamento enxuto: decisões baseadas no que é observado e foco no essencial → revise o capítulo 2

Por quê: O Scrum Guide 2020 define que o Scrum se baseia no empirismo (o conhecimento vem da experiência e as decisões se baseiam no que é observado) e no pensamento enxuto (reduzir desperdício e focar no essencial).

✗ A) Incorreta. O Scrum não se baseia em métricas individuais comparadas. O foco é o resultado do Scrum Team como um todo e o valor entregue.

✗ C) Incorreta. Um plano detalhado e fixo no início caracteriza uma abordagem preditiva. No Scrum, o plano evolui a cada Sprint com base em inspeção e adaptação.

✗ D) Incorreta. O Product Owner maximiza valor e gerencia o Product Backlog, mas não distribui tarefas. Os Developers se auto-gerenciam para decidir como fazer o trabalho.

> **Dica de prova:** Empirismo se apoia em transparência, inspeção e adaptação. Plano completo e fixo no início indica abordagem preditiva, não Scrum.

D7. C) O Scrum Team ajusta a Meta da Sprint em conjunto → revise o capítulo 3

Por quê: Na Sprint Planning, o Product Owner propõe como o produto pode aumentar seu valor e o Scrum Team colabora para definir a Meta da Sprint. Os Developers selecionam o que é possível.

✗ A) Incorreta. A Sprint Planning tem timebox e não é adiada. A meta deve ser definida antes do fim do evento.

✗ B) Incorreta. Impor uma meta inviável prejudica o compromisso e a qualidade.

✗ D) Incorreta. O Scrum Master não decide a viabilidade da meta.

> **Dica de prova:** Meta da Sprint é colaborativa: proposta do Product Owner, viabilidade dos Developers.

D8. A) A Sprint Planning perde a proposta de valor e a conversa sobre os itens → revise o capítulo 3

Por quê: A Sprint Planning é feita pelo Scrum Team inteiro. O Product Owner garante que os participantes estejam preparados e propõe como o produto pode aumentar seu valor na Sprint.

✗ B) Incorreta. O Scrum Master não assume a ordem do Product Backlog.

✗ C) Incorreta. A ausência do Product Owner não altera a duração da Sprint.

✗ D) Incorreta. A Sprint Planning é um evento de todo o Scrum Team, incluindo o Product Owner.

> **Dica de prova:** Confiança nos Developers não dispensa o Product Owner da Sprint Planning.

D9. C) Não; ela é criada pelo Scrum Team ou vem do padrão da organização → revise o capítulo 3

Por quê: Se a Definição de Pronto faz parte dos padrões da organização, todos os Scrum Teams devem segui-la como mínimo. Se não faz, o Scrum Team cria uma Definição de Pronto adequada ao produto.

✗ A) Incorreta. Comunicar depois não torna adequada uma mudança feita sozinho.

✗ B) Incorreta. O Scrum Master também não altera a Definição de Pronto sozinho.

✗ D) Incorreta. A Definição de Pronto é o compromisso do Incremento e não é responsabilidade exclusiva do Product Owner.

> **Dica de prova:** Definição de Pronto: organização ou Scrum Team. Nunca uma decisão isolada do Product Owner.

D10. B) Incluir os critérios regulatórios na Definição de Pronto → revise o capítulo 4

Por quê: A Definição de Pronto é uma descrição formal do estado do Incremento quando ele atende às medidas de qualidade exigidas para o produto. Exigências regulatórias entram nela para que cada Incremento esteja em conformidade.

✗ A) Incorreta. O roadmap não garante qualidade. As exigências precisam estar na Definição de Pronto.

✗ C) Incorreta. Sprint de conformidade não existe no Scrum. Cada Incremento deve atender à Definição de Pronto.

✗ D) Incorreta. A Daily Scrum é um evento dos Developers e não é momento de aprovação por auditor.

> **Dica de prova:** Qualidade, segurança e conformidade entram na Definição de Pronto, e não em fases posteriores.

D11. C) Avaliar Sprints mais curtas ou entregas antes do fim da Sprint → revise o capítulo 4

Por quê: Sprints mais curtas geram mais ciclos de aprendizado e limitam o risco. Além disso, Incrementos podem ser entregues aos stakeholders antes do fim da Sprint.

✗ A) Incorreta. Criar eventos extras de review não é a forma prevista pelo Scrum e adiciona sobrecarga.

✗ B) Incorreta. Relatórios não substituem a inspeção do Incremento.

✗ D) Incorreta. A duração da Sprint pode ser ajustada pelo Scrum Team entre Sprints, respeitando o limite de um mês.

> **Dica de prova:** Precisa de feedback mais frequente? Sprints mais curtas ou entregas contínuas.

D12. B) Criar expectativas irreais e pressão sobre a qualidade → revise o capítulo 4

Por quê: Previsões devem se basear em dados e no conhecimento de quem faz o trabalho. A qualidade não diminui, e promessas irreais colocam isso em risco.

✗ A) Incorreta. A necessidade de refinamento não diminui.

✗ C) Incorreta. Promessas sem base reduzem a transparência.

✗ D) Incorreta. Promessas impostas enfraquecem o auto-gerenciamento.

> **Dica de prova:** Datas prometidas sem os Developers viram pressão sobre a qualidade.

Agora, treine no formato real

Faça as 6 provas completas do curso PSPO I: 480 questões inéditas, tempo do exame, explicação de cada alternativa e link da documentação para revisar.

Ver as 6 provas completas e o cupom do mês →

Até o dia da prova, uma revisão por dia no seu feed

Instagram @strecht.simulados

Um post de prova por dia: questão comentada, pegadinhas e o "não confunda" de cada certificação.

Seguir no Instagram →

YouTube Strecht Simulados

Shorts diários com pegadinhas de prova e, aos sábados, um simulado comentado.

Inscrever-se no canal →

Material independente de preparação, sem afiliação ou endosso de Scrum.org. As questões são inéditas e não reproduzem a prova real.