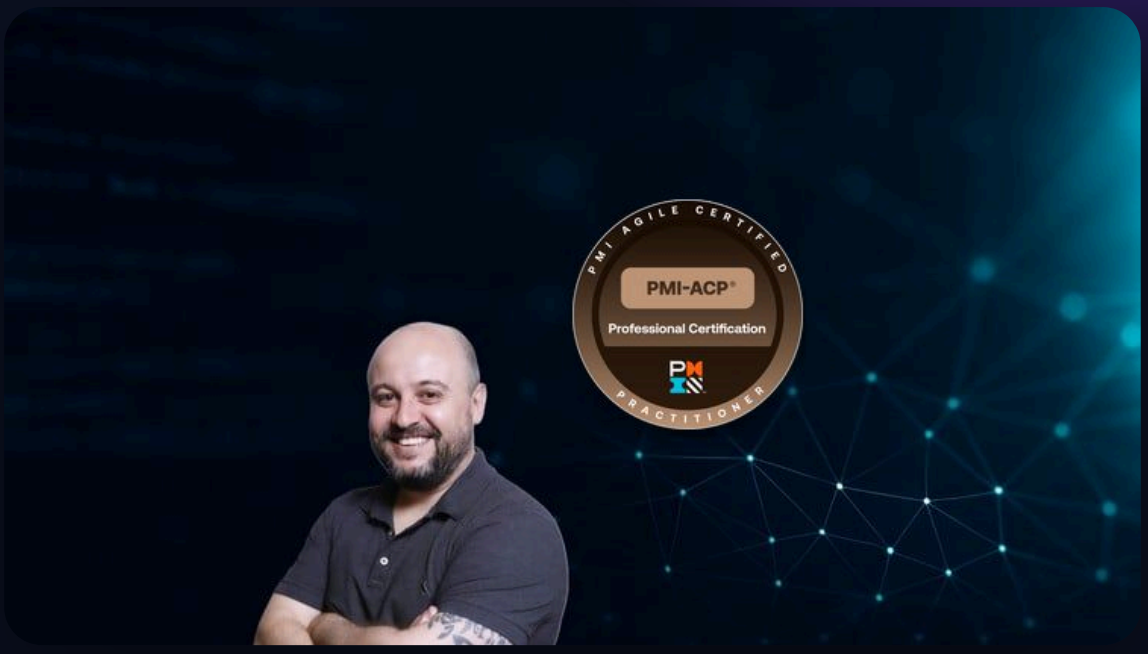
Strecht Simulados · Guia de preparação



Guia de preparação PMI-ACP

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.

4
capítulos por domínio

28
questões comentadas

72%
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 PMI-ACP: 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 72%).

Mapa do exame

Cada prova do curso tem 120 questões e 180 minutos, no formato do exame. Referência de aprovação: 72%.

Domínio (capítulo)	Peso	Neste guia
1. Domínio I. Mentalidade	28%	18 conceitos · 14 pegadinhas
2. Domínio II. Liderança	25%	18 conceitos · 14 pegadinhas
3. Domínio III. Produto	19%	18 conceitos · 14 pegadinhas
4. Domínio IV. Entrega	28%	18 conceitos · 14 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 · Domínio I. Mentalidade

Segundo o Guia do Scrum 2020, qual é o propósito e a duração da Daily Scrum?

A. Inspecionar o progresso rumo à Meta da Sprint em até 15 minutos
B. Reportar o status de cada pessoa ao Scrum Master em até 30 minutos
C. Replanejar o Product Backlog com o PO em até uma hora
D. Resolver problemas técnicos em detalhe, sem limite de tempo

D2 · Domínio I. Mentalidade

Há seis meses, o painel de um programa de modernização de uma distribuidora mostra todos os indicadores em verde. Em conversas informais, porém, os times admitem que duas integrações críticas estão atrasadas e que ninguém quer ser o primeiro a pintar o status de vermelho. O que o praticante deve fazer PRIMEIRO?

A. Criar condições para que más notícias sejam trazidas cedo, sem culpa, e refletidas no painel
B. Acrescentar ao painel novos indicadores de percentual concluído para cada uma das integrações
C. Mudar a cor padrão do painel para amarelo até que cada time comprove o próprio progresso
D. Auditar os indicadores do painel para identificar quais times informaram status incorreto

D3 · Domínio I. Mentalidade

Segundo o Guia do Scrum 2020, como deve ser tratado o refinamento do Product Backlog?

A. Como tarefa exclusiva do Scrum Master
B. Como atividade contínua, sem ser um evento
C. Como evento obrigatório com timebox de 4 horas
D. Como etapa única antes da primeira Sprint

D4 · Domínio II. Liderança

Segundo o modelo Thomas-Kilmann, qual modo de lidar com conflitos busca atender plenamente às preocupações das duas partes?

A. Evitar
B. Colaborar
C. Competir
D. Acomodar

D5 · Domínio II. Liderança

Uma secretaria estadual vai iniciar o projeto de emissão digital de licenças para pequenos comerciantes. O roteiro preparado para a reunião de abertura começa pela arquitetura técnica e pelo cronograma de marcos. O praticante sugere outra abertura, para que o time e os parceiros entendam desde o início o sentido do trabalho. Por onde a reunião deve começar?

A. Pelo problema dos comerciantes que a iniciativa resolve e pelo resultado esperado
B. Pela divisão de papéis e responsabilidades de cada área e de cada parceiro envolvido
C. Pela arquitetura de solução, já aprovada pela área de tecnologia da secretaria
D. Pelo cronograma de marcos, para dar previsibilidade aos parceiros desde o início

D6 · Domínio II. Liderança

Um time de uma companhia de saneamento perdeu dois dias porque um bloqueio no ambiente de homologação foi tratado de formas diferentes: um desenvolvedor esperou a reunião diária, outro mandou mensagem ao gerente e uma terceira abriu um chamado sem avisar ninguém. Qual ação previne a repetição do problema?

A. Criar um canal de mensagens dedicado para que o time veja os bloqueios em tempo real
B. Combinar no acordo de trabalho quem acionar, em que ordem e em quanto tempo para cada tipo de bloqueio
C. Determinar que os bloqueios sejam comunicados ao gerente, que decide como tratá-los
D. Discutir o episódio na retrospectiva e lembrar o time de avisar os colegas da próxima vez

D7 · Domínio III. Produto

Em um Agile Release Train (trem) de uma seguradora, o PO deixa no fim do backlog os habilitadores propostos pelos arquitetos, como atualizar a plataforma de mensageria e criar um ambiente de testes automatizados. Nas últimas iterações, cada funcionalidade nova levou mais tempo, e os times culpam a plataforma desatualizada. Qual deve ser a abordagem para os habilitadores?

A. Embutir os habilitadores nas estimativas das funcionalidades, sem registrá-los como itens próprios
B. Priorizar os habilitadores com as funcionalidades, por critério econômico e capacidade reservada
C. Deixar os habilitadores para uma iteração de estabilização no fim de cada incremento do programa
D. Criar um backlog técnico separado, conduzido pelos arquitetos, para os habilitadores do trem

D8 · Domínio III. Produto

Há três iterações, um time de uma distribuidora de bebidas conclui só metade do que planeja. Nas conversas, os desenvolvedores contam que passam boa parte do dia atendendo pedidos da área de operações feitos por telefone e mensagem, que não aparecem em lugar nenhum. A gerência acha que o time está lento. Qual deve ser o PRIMEIRO passo?

A. Orientar operações a abrir chamados formais e aguardar a próxima iteração para ser atendida
B. Reduzir pela metade o compromisso das próximas iterações para absorver as interrupções
C. Registrar cada pedido não planejado como cartão no quadro, com tipo próprio
D. Designar um desenvolvedor por iteração para atender os pedidos diretos da área de operações

D9 · Domínio III. Produto

No refinamento da história 'agendar uma transferência para data futura' do aplicativo de um banco, a conversa não para de crescer: e se o saldo for insuficiente no dia? E se a data cair em feriado? E se a conta de destino for encerrada antes? Após 40 minutos, ninguém consegue estimar o item. Qual deve ser o próximo passo do praticante?

A. Separar o tratamento de erros em uma história técnica a ser feita no fim do release
B. Registrar a versão mais simples como história e criar uma história para cada exceção
C. Levar a história à planning só depois de mapear cada exceção, para estimá-la uma única vez
D. Fazer um spike de uma iteração para levantar as exceções antes de dividir a história

D10 · Domínio IV. Entrega

O time de um projeto ágil com alto risco regulatório quer mostrar aos patrocinadores, a cada iteração, se a exposição total a riscos está diminuindo ao longo do tempo. Qual ferramenta atende melhor a essa necessidade?

A. Gráfico de burndown de riscos
B. Matriz de probabilidade e impacto
C. Gráfico de burnup do release
D. Diagrama de fluxo cumulativo

D11 · Domínio IV. Entrega

O aplicativo de uma operadora de planos de saúde atende beneficiários idosos, profissionais de RH das empresas clientes e corretores. O backlog mistura pedidos dos três grupos, as avaliações são contraditórias e a nota média não muda há um ano. A Product Owner trata os usuários como um público único. Qual deve ser o PRIMEIRO passo?

A. Criar uma persona única que represente o usuário médio do aplicativo da operadora
B. Priorizar os pedidos que aparecem mais vezes, somando as solicitações dos três grupos
C. Analisar cada segmento separadamente, levantando as necessidades e os resultados esperados por ele
D. Pedir à área comercial que descreva as necessidades das empresas clientes, dos corretores e dos idosos

D12 · Domínio IV. Entrega

Antes de iniciar uma release de alto risco, o time se reúne, imagina que o projeto já fracassou e lista as possíveis razões do fracasso para agir sobre elas desde já. Que técnica é essa?

A. Retrospectiva de release
B. Análise pre-mortem
C. Revisão pós-implantação
D. Análise de valor agregado

01

CAPÍTULO 1 · 28% DA PROVA

Domínio I. Mentalidade

Aqui eu preciso pensar como agilista de verdade: valores e princípios do Manifesto, empirismo e valores do Scrum, Lean, XP e Kanban, e principalmente saber escolher a abordagem certa para cada contexto (Cynefin, Stacey, filtro de adequação, híbrido). É o domínio de maior peso, junto com Entrega (28%), e mistura itens conceituais curtos (qual sequência do Cynefin, qual valor do XP, qual pilar do Scrum) com cenários longos de time, cultura e mudança organizacional. As tarefas cobradas são experimentar antes, adotar a mentalidade ágil, promover colaboração, criar transparência, promover segurança psicológica, reduzir os loops de feedback e adotar a mudança. Onde mais erro: escolher a alternativa 'organizada' (mais documento, mais aprovação, mais controle, cortar a retrospectiva) quando a resposta ágil é conversar, experimentar pequeno, tornar visível e adaptar.

Neste capítulo

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

Tópicos-chave

Cynefin (quatro domínios e suas sequências)

Claro: perceber, categorizar e responder com a boa prática (rotina estável, causa e efeito evidentes). Complicado: perceber, analisar com especialistas e responder (a causa pode ser descoberta por análise). Complexo: sondar, perceber e responder, com experimentos seguros para falhar; é onde o ágil é mais indicado. Caótico: agir primeiro para estabilizar (reverter, isolar) e só depois perceber e responder.

Matriz de Stacey

Cruza o acordo sobre requisitos com a certeza tecnológica. Simples: requisitos acordados e tecnologia conhecida. Complicada: perto de um dos eixos, como requisitos acordados e tecnologia incerta, que se ataca com spikes e protótipos antes de se comprometer. Complexa: os dois incertos, zona típica do ágil. Longe do acordo e da certeza é anarquia ou caos: reduza as duas incertezas antes de iniciar entregas regulares. 'Estável' não é zona da matriz.

Filtro (ferramenta) de adequação ágil

Avalia cultura, equipe e projeto para indicar a abordagem. Baixa incerteza e entrega única apontam que a abordagem preditiva pode ser a escolha certa; resultados mistos apontam híbrido; criticidade alta pede mais rigor de verificação e documentação, sem abandonar entregas incrementais. Cultura e confiança desfavoráveis pedem trabalhar o patrocínio e começar por um piloto antes de escalar.

Abordagem híbrida

Parte estável e de prazo longo (como hardware) segue plano preditivo; parte incerta (como o aplicativo) segue ciclos adaptativos, e as duas se sincronizam por marcos comuns. Water-Scrum-Fall é o padrão com sprints no meio e cascata nas pontas. Não existe framework único para tudo: escolha pela natureza do trabalho de cada área.

Manifesto e princípios ágeis

Os itens da direita têm valor, mas os da esquerda valem mais: é 'mais que', não 'em vez de'. Princípios mais cobrados: satisfazer o cliente com entrega cedo e contínua; mudanças tardias são bem-vindas; entregar de poucas semanas a poucos meses, preferindo o prazo mais curto; negócio e desenvolvedores juntos diariamente; projetos em torno de indivíduos motivados; conversa face a face; software funcionando como medida de progresso; excelência técnica; simplicidade (maximizar o trabalho não realizado); equipes auto-organizáveis.

Scrum: pilares, valores e compromissos

O Scrum se apoia no empirismo e no pensamento Lean. Pilares: transparência, inspeção e adaptação. Valores: compromisso, foco, abertura, respeito e coragem. Compromissos: Product Backlog tem a Meta do Produto, Sprint Backlog tem a Meta da Sprint, Incremento tem a Definição de Pronto.

Eventos do Scrum

Sprint de um mês ou menos, com duração fixa. Daily de até 15 minutos, dos Developers, para inspecionar o progresso rumo à Meta da Sprint; o formato é livre. Sprint Planning responde por que, o que e como, e o como é decidido pelos Developers. Sprint Review inspeciona o resultado e adapta o Product Backlog com os stakeholders, e não é portão de liberação. Refinamento é atividade contínua, não evento. Cancelar a Sprint é só do PO, quando a Meta da Sprint fica obsoleta.

XP: valores e práticas

Valores: comunicação, simplicidade, feedback, coragem e respeito. Práticas cobradas: propriedade coletiva do código (qualquer pessoa pode melhorá-lo), ritmo sustentável (hora extra prolongada vira defeito), refatoração (muda a estrutura, não o comportamento), releases pequenas e frequentes e YAGNI (não construa hoje o que talvez precise amanhã).

Lean e melhoria contínua

Otimizar o todo, não as partes: filas entre áreas revelam subotimização. Decidir no último momento responsável, com mais informação. Construir o que ninguém usa é desperdício de funcionalidades extras. Kaizen é melhoria em pequenos passos; kaikaku é mudança radical; Toyota Way é melhoria contínua e respeito pelas pessoas; kata de melhoria é condição-alvo, condição atual, obstáculos e experimentos; PDCA é planejar, fazer, checar e agir.

Lean Startup e experimentação

Ciclo construir, medir, aprender, seguido da decisão de pivotar ou perseverar. Métrica, limiar e decisão se definem antes do teste; resultado negativo com critério definido é aprendizado validado. Teste primeiro a hipótese mais arriscada, normalmente se o cliente quer a solução. Prefira evidência de comportamento (usar, pagar) a opinião declarada, e métricas acionáveis por coorte a totais acumulados de vaidade.

Mágico de Oz, concierge e teste de fumaça

Mágico de Oz: interface real com operação humana escondida nos bastidores. Concierge: serviço feito à mão e às claras para poucos clientes. Teste de fumaça mede interesse com um clique, e o usuário deve ser avisado com clareza, logo após o clique, que o serviço está em teste e nada será cobrado. Para validar demanda com pouco dinheiro, atender poucos clientes à mão cobrando uma taxa simbólica gera evidência de comportamento.

Design thinking

Comece pela empatia: observar e entrevistar quem vive o problema quando o time pulou direto para a solução. Muitas entrevistas sem padrão se sintetizam em um mapa de empatia (o que dizem, pensam, fazem e sentem). Processo que atravessa vários canais pede mapa da jornada do cliente com emoções e pontos de atrito. Divergir antes de convergir: muitas ideias primeiro, julgamento depois, e protótipos rápidos para testar.

Outros métodos ágeis

FDD: modelo geral, lista de funcionalidades e construção por funcionalidade. DSDM: tempo, custo e qualidade fixos, escopo variável com MoSCoW. Crystal: cerimônia proporcional ao tamanho do time e à criticidade. Kanban: comece com o que você faz hoje e evolua aos poucos. Scrumban: cadência do Scrum com fluxo, limites de WIP e classes de serviço, para trabalho planejado misturado com demanda imprevisível.

Coordenação entre times

O mecanismo é dimensionado pelo número de dependências e pela frequência de integração, não pela moda. LeSS: um produto, um PO, um backlog e uma Sprint comum. Um produto com vários times e incremento integrado pede time de times com backlog único, cadência comum, eventos sincronizados e uma Definição de Pronto comum. Scrum of Scrums trata dependências e impedimentos, com representantes em rodízio, e não é relatório de status ao gerente.

Modelo Spotify

Squad é o time; tribo agrupa squads; chapter reúne pessoas da mesma especialidade; guilda reúne pessoas por interesse. A prova cobra a diferença entre chapter (especialidade) e guilda (interesse).

Segurança psicológica e cultura

Diretiva Primária da retrospectiva: foco no aprendizado, sem culpa. Westrum classifica culturas em patológica, burocrática e generativa; o ágil busca a generativa. McGregor: a Teoria X controla, a Teoria Y confia, e o ágil parte da Y. Segurança alta com padrão baixo é zona de conforto: a meta é segurança alta com exigência alta. Para o time admitir erros, a liderança admite os próprios primeiro.

Modelos de mudança organizacional

Kotter: tudo começa com o senso de urgência. ADKAR: consciência, desejo, conhecimento, habilidade e reforço; identifique o primeiro elemento que falta (gente treinada que não adota costuma ter problema de desejo). Curva de Satir: a mudança traz uma queda temporária (caos) antes do ganho. Dweck: a mentalidade fixa trava e a de crescimento aprende; elogie o processo (esforço e estratégia), não o talento.

Tuckman e estabilidade do time

Estágios: formação, conflito, normatização, desempenho e dissolução. Quando o time cria as próprias normas (normatização), o líder reduz a direção e passa a apoiar. Mudou a composição do time, espere a volta a estágios anteriores e reintegre com pareamento e revisão dos acordos. Leve o trabalho até times estáveis e de longa duração: remontar times a cada projeto repete o custo de formação.

Macetes para memorizar

Sequências do Cynefin

Claro CAtegoriza, Complicado ANAlisa, Complexo SONDA, Caos AGE: o verbo do meio muda, o 'responder' fica sempre no fim.

Pilares x valores do Scrum

Pilares são 3 (TIA: Transparência, Inspeção, Adaptação); valores são 5 (Compromisso, Foco, Abertura, Respeito, Coragem).

Valores do XP

XP tem Coragem e Respeito como o Scrum, mas troca o resto por Comunicação, Simplicidade e Feedback.

Compromissos do Scrum

Cada artefato tem sua meta: Product Backlog -> Meta do Produto; Sprint Backlog -> Meta da Sprint; Incremento -> Definição de Pronto.

Spotify

Do menor ao maior: squad (o time) dentro da tribo (grupo de squads). Chapter = mesma ESPECIALIDADE, dentro da tribo; guilda = mesmo INTERESSE, aberta a toda a empresa.

Cultura

Teoria X = X de 'xerife' (controla); Y = confia. Westrum: o ágil quer a Generativa.

Mudança

Kotter começa pela urgência; ADKAR começa pela consciência e trava muito no desejo; Satir sempre passa pelo caos antes de melhorar.

Tabela de memorização: não confunda

Quando é...	...e quando é
Cynefin complicado (perceber, analisar, responder) Especialistas conseguem chegar à causa analisando logs, dados e planos; há várias respostas boas e tempo para analisar.	**Cynefin complexo (sondar, perceber, responder)** Causa e efeito só ficam claros em retrospecto; use várias sondagens pequenas e seguras para falhar, em paralelo, e amplie o que funcionar.
Cynefin caótico (agir, perceber, responder) Dano em curso e ninguém entende a causa: estabilize primeiro (reverter, isolar), investigue depois.	**Cynefin claro (perceber, categorizar, responder)** Rotina estável, documentada e com causa e efeito evidentes: siga o procedimento padrão, sem experimentar variações.
Iterativo Refina a solução em ciclos sucessivos, melhorando a mesma coisa com feedback.	**Incremental** Entrega partes utilizáveis em sequência; requisitos estáveis com cliente querendo usar partes cedo pedem ciclo incremental.
Kaizen Melhoria contínua em pequenos passos, feita no dia a dia.	**Kaikaku** Mudança radical, de ruptura, em vez de ajustes graduais.
Radiador de informação Informação visível sem esforço, como quadro e burnup na parede ou painel aberto.	**Refrigerador de informação** Informação guardada e difícil de achar, como Definição de Pronto em uma pasta ou relatório que ninguém abre.
Mágico de Oz O cliente usa uma interface que parece real, mas a operação é feita por pessoas escondidas nos bastidores.	**Concierge** O serviço é feito à mão e às claras, para poucos clientes, sem fingir automação.

Spike (timebox técnico) A dúvida é técnica, como se um motor ou tecnologia atinge o desempenho exigido.	**Experimento com clientes** A dúvida é de mercado, como se clientes aceitam a taxa ou querem a solução; com as duas dúvidas, rode os dois.

Números que caem

Item	Valor
Peso do domínio na prova	**28%**
Duração máxima da Sprint	**Um mês ou menos, com duração fixa**
Duração da Daily Scrum	**Até 15 minutos**
Frequência de entrega pelo Manifesto	**De poucas semanas a poucos meses, preferindo o prazo mais curto**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Analisar com especialistas antes de mexer no sistema quando há dano em curso e risco crescente.
Como não cair: Analisar antes de agir é do domínio complicado; no caos o tempo de análise prolonga a exposição. Primeiro estabilize, depois investigue.

⚠ **Alerta de prova:** Testar pequenas variações de um procedimento mensal estável porque 'experimentar é ágil'.
Como não cair: Sondar é para o complexo; numa rotina documentada com causa e efeito evidentes, variar o procedimento só introduz risco. Domínio claro: boa prática.

⚠ **Alerta de prova:** Com requisitos acordados e tecnologia incerta, marcar zona complexa e voltar a rever os requisitos com as áreas.
Como não cair: Na Stacey, isso é zona complicada; o problema está na tecnologia, então reduza a incerteza técnica com spikes e protótipos.

⚠ **Alerta de prova:** Achar que o ágil sempre vence e descartar a abordagem preditiva.
Como não cair: Quando o filtro de adequação mostra baixa incerteza e entrega única, a abordagem preditiva pode ser a escolha certa.

⚠ **Alerta de prova:** Ler o Manifesto como 'em vez de': documentação, contratos e planos não têm valor.
Como não cair: Os itens da direita têm valor; os da esquerda valem mais. É 'mais que'.

⚠ **Alerta de prova:** Sob pressão de prazo, cancelar a retrospectiva para ganhar tempo.
Como não cair: Pressão de prazo não justifica cortar a retrospectiva: encurte e foque nos riscos da entrega, não elimine.

⚠ **Alerta de prova:** Na retrospectiva, sair com uma lista longa de melhorias genéricas como 'melhorar a revisão'.
Como não cair: Retrospectiva eficaz gera uma ou duas ações com dono, no backlog da próxima iteração, com hipótese, prazo e métrica, e verifica o resultado depois.

⚠ **Alerta de prova:** Depois de um incidente, procurar quem errou ou acrescentar mais aprovações.
Como não cair: Pós-incidente sem culpa: reconstruir a linha do tempo e as condições do sistema; automatizar a verificação em vez de empilhar aprovações.

⚠ **Alerta de prova:** Quando o resultado de um experimento sai negativo, tratar como fracasso e procurar o culpado.
Como não cair: Com critério definido antes, resultado negativo é aprendizado validado: mostre o que o teste ensinou, o limite de perda previsto e decida pivotar ou perseverar.

⚠ **Alerta de prova:** Reagir a um pedido de mudança tardia e obrigatória recusando ou adiando para o fim do projeto.
Como não cair: Mudanças tardias são bem-vindas se trazem valor: acolha, avalie o impacto com a PO e reordene o backlog das iterações restantes.

⚠ **Alerta de prova:** Mudança de prioridade no meio da iteração: interromper tudo e trocar o trabalho na hora.
Como não cair: Se a meta ainda vale, proteja-a; avalie o impacto com a PO e ordene a resposta no backlog para a próxima iteração. Só o PO cancela a Sprint, e só se a meta ficar obsoleta.

⚠ **Alerta de prova:** Dividir um time grande por especialidade ou componente (front, back, banco).
Como não cair: Dependências em cada história indicam times por componente; divida por funcionalidade (fluxo de valor), com times multifuncionais que entregam de ponta a ponta.

⚠ **Alerta de prova:** Ver um herói sobrecarregado como solução e manter tudo com ele.
Como não cair: Herói à beira do esgotamento é risco do time: propriedade coletiva, pareamento e rodízio nos plantões e tarefas críticas.

⚠ **Alerta de prova:** Usar o resultado do ESVP ou de uma pesquisa anônima com resultado ruim para descobrir quem respondeu o quê.
Como não cair: São termômetros anônimos: discuta o resultado com o grupo e nunca tente identificar pessoas; tentar descobrir quem respondeu viola o anonimato que torna a resposta honesta.

Como cai na prova

📝 **Como cai:** Cenário descreve um problema (incidente, migração, rotina) e pergunta qual domínio do Cynefin ou qual sequência aplicar.
Como resolver: Pergunte: há dano em curso? (caótico). Especialistas descobrem a causa analisando? (complicado). Só dá para saber testando? (complexo). Rotina documentada? (claro). Depois descarte as alternativas pelo verbo da sequência.

📝 **Como cai:** Organização quer escolher abordagem (ágil, preditiva, híbrida) com base no filtro de adequação ou nas características do projeto.
Como resolver: Leia o que é estável e o que é incerto. Tudo estável e entrega única: preditivo. Tudo incerto: ágil. Misto: híbrido sincronizado. Cultura desfavorável: patrocínio e piloto antes de escalar.

📝 **Como cai:** Time ou líder enfrenta problema cultural (medo, culpa, status verde, ninguém fala) e pede a melhor ação.
Como resolver: Prefira a alternativa que cria segurança e transparência sem culpa (agradecer o alerta, linha do tempo, conversa no time). Descarte punir, procurar culpado, esconder do stakeholder ou escalar de cara.

📝 **Como cai:** Proposta de inovação ou ideia nova com pouco tempo ou pouco dinheiro.
Como resolver: Procure a menor fatia testável com usuários reais, com hipótese, métrica, limiar, timebox e critério de parada definidos antes. Descarte construir a versão completa ou pesquisar só opinião.

📝 **Como cai:** Item conceitual curto: 'segundo o Guia do Scrum / o Manifesto / o XP, qual...'.
Como resolver: Responda com o termo exato da fonte (ex.: Sprint de um mês ou menos; Daily de 15 minutos dos Developers; 'mais que' no Manifesto) e desconfie de alternativas que trocam papéis ou eventos.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
Causa e efeito só ficam claros em retrospecto	**Cynefin complexo: sondar, perceber, responder**

Sistema causando dano agora e ninguém sabe a causa	**Caótico: agir para estabilizar, depois investigar**
Requisitos acordados, tecnologia nunca usada	**Stacey zona complicada: spikes e protótipos**
Requisitos e tecnologia incertos	**Stacey zona complexa: abordagem ágil**
Parte estável de prazo longo e parte incerta no mesmo programa	**Híbrido sincronizado por marcos comuns**
Sprints no meio, cascata no início e no fim	**Water-Scrum-Fall**
Status sempre verde e problemas escondidos	**Tornar seguro trazer más notícias cedo, sem culpa**
Treinados que não adotam a mudança	**ADKAR: falta o desejo**
Queda de desempenho logo após introduzir a mudança	**Curva de Satir (fase de caos)**
Time cria as próprias regras e resolve divergências sozinho	**Tuckman normatização: líder reduz direção e apoia**
Demanda imprevisível com prazos variados	**Fluxo contínuo puxado com limites de WIP e classes de serviço (ou Scrumban)**
Mercado volátil e cadência longa	**Iterações mais curtas, com eventos reduzidos proporcionalmente**
Pessoas repetem a opinião de quem falou primeiro	**Escrita silenciosa individual antes da discussão**
Fusos muito distantes no mesmo time	**Status assíncrono por escrito e encontro curto em horário que se alterna**

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

- Complexo = sondar, perceber, responder; é o domínio onde o ágil brilha. Caos = agir primeiro.
- O ágil não é sempre a resposta: baixa incerteza e entrega única podem pedir abordagem preditiva.
- Sem culpa: falha se analisa pelas condições do sistema, e más notícias cedo merecem agradecimento.
- Experimento bom tem hipótese, métrica, limiar, timebox e decisão combinados antes de começar.
- Só o PO cancela a Sprint, e só quando a Meta da Sprint fica obsoleta.

Essenciais para a prova (checklist)

- ☐ Sei a sequência de resposta de cada domínio do Cynefin e reconheço cada um em um cenário.
- ☐ Sei posicionar um projeto na matriz de Stacey pelos eixos acordo e certeza tecnológica.
- ☐ Sei interpretar o resultado de um filtro de adequação (preditivo, híbrido, ágil, piloto).
- ☐ Sei os pilares, valores, eventos e compromissos do Scrum com seus timeboxes.
- ☐ Sei os valores do XP e as práticas propriedade coletiva, ritmo sustentável, refatoração e YAGNI.
- ☐ Sei diferenciar kaizen, kaikaku, PDCA e kata de melhoria.
- ☐ Sei diferenciar Mágico de Oz, concierge, teste de fumaça e spike.
- ☐ Sei aplicar Kotter, ADKAR, Satir, Tuckman, Westrum e Teoria X/Y a cenários.
- ☐ Sei escolher o mecanismo de coordenação entre times (LeSS, time de times, Scrum of Scrums) pelas dependências.
- ☐ Sei montar uma retrospectiva que gera poucas ações com dono e verificação na iteração seguinte.

Questões de revisão do capítulo

R1.1 · Domínio I. Mentalidade

Em qual situação uma abordagem híbrida, que combina práticas preditivas e ágeis, tende a ser a mais adequada?

A. Projeto com partes de requisitos estáveis e partes incertas
B. Equipe pequena e colocalizada criando um aplicativo inédito
C. Obra repetitiva com escopo conhecido e muito estável
D. Produto novo com requisitos pouco conhecidos e mutáveis

R1.2 · Domínio I. Mentalidade

Faltando duas iterações para o lançamento de um aplicativo de crédito consignado, o órgão regulador publica uma norma que altera o cálculo das taxas exibidas ao cliente. O time tem boa cobertura de testes automatizados e integração contínua. Como o time deve tratar a mudança?

A. Acolher a mudança, avaliar o impacto com a PO e reordenar o backlog das iterações restantes
B. Adiar o lançamento até que o jurídico produza a especificação completa da nova norma
C. Registrar a mudança para a versão seguinte, preservando o escopo combinado para o lançamento
D. Abrir um pedido formal de mudança de escopo para o comitê aprovar antes de analisá-la

R1.3 · Domínio I. Mentalidade

Uma fabricante de equipamentos médicos precisa definir o sistema de bateria de um novo monitor portátil. Há três tecnologias candidatas, e os testes de autonomia e aquecimento só terão resultados confiáveis em oito semanas. Escolher a errada agora custaria meses de retrabalho no gabinete e no software. Qual abordagem é a MAIS adequada?

A. Desenvolver as três opções em paralelo até os testes e descartar as que falharem nos limites definidos
B. Escolher a tecnologia mais bem avaliada pelos especialistas e seguir só com ela até os testes
C. Adiar o início do gabinete e do software até que os testes de bateria sejam concluídos
D. Congelar os requisitos de bateria agora para que o projeto do gabinete possa avançar

R1.4 · Domínio I. Mentalidade

Uma companhia de seguros definiu três objetivos estratégicos para o ano, como reduzir em 20% o prazo de regulação de sinistros. Ao revisar os backlogs de oito times, o praticante percebe que a maioria dos itens vem de pedidos diretos das áreas e não tem relação clara com esses objetivos. O que ele deve propor?

A. Desdobrar os objetivos em resultados mensuráveis por time e usá-los como critério de priorização
B. Pedir a cada time que acrescente em cada item uma etiqueta com o objetivo relacionado
C. Aumentar a capacidade dos times para atender aos pedidos e aos objetivos ao mesmo tempo
D. Encaminhar os pedidos diretos das áreas a um comitê que aprove cada item novo do backlog

Respostas comentadas

R1.1. A) Projeto com partes de requisitos estáveis e partes incertas

Por quê: Abordagens híbridas combinam planejamento preditivo para componentes com requisitos estáveis, como infraestrutura ou obrigações regulatórias, com ciclos ágeis para as partes incertas, como funcionalidades de software. Requisitos incertos pedem ágil; muito estáveis, preditivo.

✗ B) Incorreta. Um aplicativo inédito com equipe pequena favorece o ágil.
✗ C) Incorreta. Escopo conhecido e estável favorece o preditivo.
✗ D) Incorreta. Requisitos incertos e mutáveis favorecem o ágil.

> **Dica de prova:** Parte estável mais parte incerta: abordagem híbrida.

R1.2. A) Acolher a mudança, avaliar o impacto com a PO e reordenar o backlog das iterações restantes

Por quê: Um dos princípios do Manifesto Ágil é aceitar mudanças de requisitos, mesmo tarde no desenvolvimento. Práticas técnicas como testes automatizados e integração contínua mantêm baixo o custo da mudança, permitindo que o time repriorize o backlog e incorpore a nova regra com segurança.

✗ B) Incorreta. Esperar a especificação completa atrasa sem necessidade; o time pode trabalhar a mudança de forma incremental com o jurídico.

✗ C) Incorreta. Lançar com um cálculo fora da norma expõe a empresa e os clientes; preservar o escopo não justifica descumprir a regra.

✗ D) Incorreta. Um processo formal de aprovação antes da análise adiciona espera a uma mudança obrigatória que o time pode absorver.

> **Dica de prova:** Mudança tardia e obrigatória: acolha, avalie o impacto e repriorize; boas práticas técnicas barateiam a mudança.

R1.3. A) Desenvolver as três opções em paralelo até os testes e descartar as que falharem nos limites definidos

Por quê: O princípio 3 do SAFe, assumir a variabilidade e preservar opções, recomenda o projeto baseado em conjuntos quando a incerteza técnica é alta. Em vez de escolher cedo uma solução única, o time desenvolve algumas alternativas e as elimina à medida que os dados chegam. O custo de manter as opções costuma ser menor que o de um retrabalho causado por uma escolha precoce errada.

✗ B) Incorreta. É a abordagem de ponto único: escolhe cedo, no auge da incerteza, e transforma um erro provável em retrabalho caro.

✗ C) Incorreta. Parar o restante do projeto desperdiça oito semanas; é possível avançar mantendo o gabinete e o software compatíveis com as opções.

✗ D) Incorreta. Congelar requisitos não reduz a incerteza técnica; apenas a esconde até os testes revelarem o problema.

> **Dica de prova:** Alta incerteza e decisão cara de reverter: preserve opções em paralelo e decida quando houver dados.

R1.4. A) Desdobrar os objetivos em resultados mensuráveis por time e usá-los como critério de priorização

Por quê: A revisão de 2026 do ECO reforça a agilidade empresarial: alinhar continuamente o trabalho às prioridades da organização. Ferramentas como OKRs desdobram objetivos estratégicos em resultados mensuráveis por time. Esses resultados passam a orientar a priorização do backlog, reduzindo a distância entre estratégia e execução.

✗ B) Incorreta. Etiquetar itens é cosmético; não muda o que entra nem a ordem do backlog.

✗ C) Incorreta. Mais capacidade não resolve a falta de foco; os times continuariam dispersos em itens desconectados da estratégia.

✗ D) Incorreta. Um comitê para cada item cria fila e burocracia; o alinhamento deve vir de objetivos claros usados pelos próprios times.

> **Dica de prova:** Estratégia que não aparece no critério de priorização dos backlogs não chega à execução; desdobre objetivos em resultados por time.

02

CAPÍTULO 2 · 25% DA PROVA

Domínio II. Liderança

Neste domínio (25% da prova) eu preciso agir como líder servidor: empoderar o time, facilitar a resolução de problemas, promover o compartilhamento de conhecimento, os princípios ágeis, a visão compartilhada e o gerenciamento de conflitos. Os itens cobram muitos modelos com nome próprio (Speed Leas, Thomas-Kilmann, Lencioni, liderança situacional, Pink, Herzberg, GROW, Johari, escuta ativa, tipos de poder) e cenários em que preciso escolher entre ensinar, fazer coaching, dar mentoria ou deixar o time decidir. A resposta certa quase sempre devolve a decisão ao time, ataca a causa-raiz e cria confiança; as erradas decidem pelo time, escalam cedo demais ou tratam só o sintoma. Onde mais erro é no nível exato do conflito e no estilo situacional adequado à pessoa.

Neste capítulo

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

Tópicos-chave

Liderança servidora

Servir primeiro e fazer as pessoas crescerem. O líder define o quê e o porquê; o time decide o como. Time que pede para receber tarefas deve ter a decisão devolvida, com facilitação, e não distribuição de trabalho. Para o time decidir bem na ausência do líder, deixe claro o propósito e o resultado esperado, não um roteiro.

Treinamento, coaching, mentoria e consultoria

Coaching pergunta; mentoria aconselha a partir da experiência; ensino explica; consultoria resolve. Pessoa não sabe fazer: treine. Sabe, mas precisa evoluir: coaching. Precisa da experiência de alguém: mentoria. Perguntas não ajudam quem não tem a informação: ensine primeiro e depois volte ao coaching.

GROW

Roteiro de conversa de coaching: objetivo, realidade, opções e ação. Não confunda com SMART (critérios de metas), SWOT (forças, fraquezas, oportunidades, ameaças) ou OSKAR, outro modelo de coaching com etapas diferentes. Coaching com patrocinador exige acordo explícito de objetivos e sigilo antes da primeira sessão.

Liderança situacional

Estilos direcionar, orientar, apoiar e delegar conforme a maturidade da pessoa na tarefa. Novato sem competência: direcionar, com orientações claras e acompanhamento. Competente mas desmotivado: apoiar (muito apoio, pouca direção). Competente e comprometido: delegar o como e combinar o resultado. Delegar cedo demais expõe a pessoa ao fracasso.

Bases de poder

Poder de referência: influência por admiração e relacionamento, sem cargo nem expertise. Poder de especialista: influência pelo conhecimento. Legítimo vem do cargo formal (o papel de Scrum Master não dá autoridade formal sobre o time); recompensa vem de oferecer benefícios; coercitivo vem de punições ou ameaças.

Liderança transformacional x transacional

Transformacional inspira e desafia: visão, exemplo, estímulo intelectual e atenção individual. Transacional troca recompensa por resultado. Marquet propõe trocar o pedido de permissão pela declaração de intenção.

Motivação: Pink e Herzberg

Daniel Pink: autonomia, maestria e propósito (pertencimento, segurança e status lembram Maslow, não Pink). Herzberg: fatores higiênicos, como salário, evitam insatisfação, mas não motivam; motivadores (realização, reconhecimento, o próprio trabalho) geram engajamento. Profissional estagnado ganha com desafio maior e apoio, alimentando a maestria.

Reconhecimento e incentivos

Recompensa ligada a story points infla as estimativas e mata a colaboração. O que se premia se repete: reconheça a prevenção e a colaboração, não o heroísmo. Reconhecimento leve, frequente e entre pares reforça a colaboração melhor que prêmios individuais; experimento que refuta a hipótese cedo também merece palco.

Inteligência emocional e comunicação

Começa por dentro: autoconsciência e autorregulação. Escuta ativa tem três níveis: interna (nível 1, foco nos próprios pensamentos), focada e global (nível 3). Sinal não verbal pede verificação com a pessoa, não conclusão nem exposição pública. Comunicação não violenta: observação, sentimento, necessidade e pedido. Feedback eficaz: situação, comportamento observável e impacto.

Janela de Johari e escada da inferência

Na Johari, o feedback reduz a área cega e a autorrevelação reduz a área oculta. A escada da inferência explica saltos da observação à ação em segundos; para corrigir, volte os degraus até os dados observados.

Lencioni: cinco disfunções

A base é a ausência de confiança; sem ela surge o medo do conflito, depois a falta de comprometimento, a fuga da responsabilização (quarto nível) e, no topo, a desatenção aos resultados. Debate intenso sobre ideias é saudável; o risco é ele virar pessoal.

Níveis de conflito de Speed Leas

Nível 1, problema a resolver: troca aberta de informação. Nível 2, desacordo: autoproteção e linguagem cautelosa; crie segurança e leve a conversa para os interesses. Nível 3, disputa: lados, ganhar fica mais importante que resolver e surgem ataques; facilite de forma estruturada, com regras e foco em fatos. Nível 4, cruzada: defesa de princípios e ataque ao outro grupo. Nível 5, guerra mundial: intenção de destruir o outro; separe as partes e busque intervenção externa.

Modos de Thomas-Kilmann

Cinco modos: competir, colaborar, comprometer (conciliar), evitar e acomodar. Colaborar atende plenamente aos dois lados. Competir serve para emergência ou segurança: decida rápido e explique depois. Conciliar resolve rápido com prazo curto e metas de importância moderada. Acomodar é legítimo quando o tema é irrelevante e a relação é importante; evitar só adia.

Análise de causa-raiz

Cinco porquês que param em 'erro humano' ainda não chegaram à causa. Ishikawa gera hipóteses: use votação para escolher a mais provável e verifique-a primeiro; com muitas soluções e nenhum diagnóstico, Ishikawa antes de solução. A3 reúne problema, causa, contramedidas e plano em uma folha. Problema afetando clientes: contenha primeiro e investigue logo depois; ação implantada só encerra o problema depois de verificar o indicador.

Compartilhamento de conhecimento

Comunidade de prática: participação voluntária, aberta a vários times e papéis; recomenda e inspira, não impõe padrões nem vira comitê de aprovação; a liderança patrocina com tempo e os membros conduzem. Conhecimento concentrado em uma pessoa pede matriz de competências e pareamento, com rodízio planejado. Conhecimento tácito se transfere fazendo junto; compartilhar precisa de capacidade reservada e visível.

Visão e propósito compartilhados

Comece pelo porquê: problema e resultado antes do como e do quando. Caixa do produto expõe e alinha leituras diferentes do propósito. Quadro da visão: público-alvo, necessidades, produto e objetivos de negócio (prazo não é objetivo de negócio). Visão vaga se traduz em meta de produto mensurável e se retoma nos eventos de planejamento e revisão; meta da organização mudou, realinhe roadmap e métricas agora.

Técnicas de decisão em grupo

Fist of five mede rapidamente o apoio a uma proposta (zero a cinco). Delegation poker torna explícito quem decide o quê. Consentimento: a proposta avança se não houver objeção fundamentada. Discordar e se comprometer: debate aberto, decisão tomada, apoio pleno. Sem consenso e com prazo, aplique a regra de recurso já combinada; consenso rápido em decisão cara pede advogado do diabo.

Tamanho do time e comunicação

Time Scrum em geral com 10 pessoas ou menos; se cresceu, divida. Canais de comunicação = n(n-1)/2, número que cresce rápido com o tamanho. Lei de Brooks: mais gente em projeto atrasado atrasa mais. Comunicação osmótica é aprender ouvindo o time ao redor.

Macetes para memorizar

💡 Speed Leas

Problema, Desacordo, Disputa, Cruzada, Guerra: 'PDD-CG'. Até o 2 o time resolve com segurança; no 3 o facilitador estrutura; no 5 separa e chama ajuda externa.

💡 Treinar x coaching x mentoria

Não SABE: treine. Sabe e quer CRESCER: coaching. Quer a EXPERIÊNCIA de alguém: mentoria. Coach pergunta, mentor aconselha, professor explica, consultor resolve.

💡 Situacional

Direcionar -> Orientar -> Apoiar -> Delegar acompanha a pessoa de 'não sei' até 'sei e quero'. Desmotivado e competente = APOIAR.

💡 Lencioni (de baixo para cima)

Confiança, Conflito, Comprometimento, Responsabilização, Resultados: sem confiança não há conflito saudável.

💡 Matriz de stakeholders

Poder alto + interesse alto = gerenciar de perto; poder alto + interesse baixo = manter Satisfeito; poder baixo + interesse alto = manter Informado; baixo + baixo = monitorar.

💡 Herzberg

Higiênico é como higiene: se faltar incomoda, se sobrar não motiva.

💡 CNV

OSNP: Observação, Sentimento, Necessidade, Pedido.

Tabela de memorização: não confunda

Quando é...	...e quando é

Coaching A pessoa tem o conhecimento e precisa evoluir: o coach faz perguntas poderosas e não dá receita.	**Mentoria** A pessoa domina a técnica mas precisa da experiência de alguém, como o contexto de um sistema legado ou 'como você lidou com isso?'.
Liderança situacional: apoiar Pessoa competente mas desmotivada ou insegura: ouvir, encorajar e envolver nas decisões, com pouca orientação sobre a tarefa.	**Liderança situacional: delegar** Pessoa competente e comprometida: combinar o resultado e deixar o como com ela.
Poder de referência Influência por admiração e relacionamento, mesmo sem domínio técnico do assunto.	**Poder de especialista** Influência pelo conhecimento técnico, sem cargo hierárquico.
Speed Leas nível 2 (desacordo) Autoproteção e cautela, sem ataques: crie segurança e deixe que exponham interesses.	**Speed Leas nível 3 (disputa)** Lados formados, vencer importa mais que resolver e há ataques pessoais: facilitação estruturada com regras e foco em fatos.
Thomas-Kilmann: competir Emergência ou segurança em jogo: decidir rápido e explicar depois.	**Thomas-Kilmann: acomodar** Tema irrelevante para você e relação importante: ceder é escolha legítima.
Transformacional Inspira com visão, exemplo, estímulo intelectual e atenção individual.	**Transacional** Troca recompensa por resultado.
Fist of five Medir rapidamente o apoio do grupo a uma proposta, em escala de zero a cinco.	**Delegation poker** Definir o nível de autoridade entre líder e time para cada tipo de decisão.

Números que caem

Item	Valor
Peso do domínio na prova	**25%**
Tamanho recomendado do time Scrum	**Em geral 10 pessoas ou menos**
Canais de comunicação	**n(n-1)/2; time de 9 pessoas = 36 canais**
Níveis de conflito de Speed Leas	**5 (problema, desacordo, disputa, cruzada, guerra mundial)**
Modos de Thomas-Kilmann	**5 (competir, colaborar, comprometer, evitar, acomodar)**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Fazer coaching com perguntas para alguém que simplesmente não tem a informação.
Como não cair: Perguntas não ajudam quem não sabe: ensine o essencial primeiro, faça junto e retire o apoio aos poucos, depois volte ao coaching.

⚠ **Alerta de prova:** Delegar logo a um novato motivado para mostrar confiança.
Como não cair: Delegar cedo expõe ao fracasso; quem ainda não tem competência pede direcionar, com orientação clara e acompanhamento.

⚠ **Alerta de prova:** Direcionar de perto uma especialista que perdeu a motivação.
Como não cair: A lacuna é de motivação, não de conhecimento: estilo apoiar. Direcionar soa como desconfiança e delegar reduz o contato que ela precisa.

⚠ **Alerta de prova:** Achar que o papel de Scrum Master dá poder legítimo sobre o time.
Como não cair: O Scrum Master não tem autoridade formal; influência sem cargo vem de referência (admiração) ou de especialista (conhecimento).

⚠ **Alerta de prova:** Esperar que um aumento salarial resolva a falta de engajamento.
Como não cair: Salário é fator higiênico: evita insatisfação, não motiva. Motivadores são realização, reconhecimento e o próprio trabalho.

⚠ **Alerta de prova:** Premiar times pela velocidade ou por story points para estimular desempenho.
Como não cair: Recompensa ligada a pontos infla estimativas e mata a colaboração; meça fluxo e resultado, não ocupação nem pontos.

⚠ **Alerta de prova:** Escalar ao gerente um desacordo sem ataques pessoais.
Como não cair: Nível 2 ainda se trata no time: crie segurança para expor interesses. Escalar é desproporcional e tira das partes a chance de resolver.

⚠ **Alerta de prova:** Decidir a questão técnica pelos dois lados para encerrar o desgaste.
Como não cair: O facilitador cuida do processo; a decisão técnica é do time. Se as partes tentaram e não conseguiram, medie sem decidir por elas.

⚠ **Alerta de prova:** Pedir ao Product Owner que escolha uma biblioteca técnica quando o time não chega a consenso.
Como não cair: O PO responde pelo valor, não por escolhas técnicas; aplique a regra de decisão de recurso já combinada pelo time.

⚠ **Alerta de prova:** Encerrar a análise de causa-raiz em 'erro humano'.
Como não cair: Cinco porquês que param em erro humano ainda não chegaram à causa-raiz; continue até a condição do sistema.

⚠ **Alerta de prova:** Transformar a comunidade de prática em comitê que aprova padrões.
Como não cair: Comunidade de prática é voluntária, aberta e recomenda; a liderança patrocina com tempo, os membros conduzem.

⚠ **Alerta de prova:** Designar uma única pessoa fixa para integrar novatos ou manter pares fixos para sempre.
Como não cair: Concentrar cria novo gargalo e nova ilha de conhecimento: guia de integração do time e rodízio planejado de pares.

⚠ **Alerta de prova:** Usar a média de um health check ou autoavaliação como meta ou ranking entre times.
Como não cair: Autoavaliação é do time: olhe a tendência do próprio time, discuta notas dispersas antes de agir sobre a média e escolha um foco por vez.

⚠ **Alerta de prova:** Colocar mais gente em um projeto atrasado para recuperar o prazo.
Como não cair: Lei de Brooks: mais gente em projeto atrasado atrasa mais; os canais de comunicação crescem por n(n-1)/2.

Como cai na prova

📝 **Como cai:** Cenário de conflito com sinais de comportamento (cautela, lados, ataques) e pergunta o nível ou a ação.
Como resolver: Classifique pelo comportamento: troca aberta (1), autoproteção (2), vencer e ataques (3), princípios e ataque ao grupo (4), destruir (5). Escolha a intervenção proporcional ao nível; descarte escalar cedo ou separar as partes antes do 5.

📝 **Como cai:** Membro do time com determinada competência e motivação; qual estilo ou abordagem de desenvolvimento usar.
Como resolver: Separe competência de motivação. Falta competência: direcionar ou treinar. Falta motivação: apoiar. Tem os dois: delegar. Falta contexto de alguém experiente: mentoria.

📝 **Como cai:** Time pede ao líder que decida, distribua tarefas ou resolva o problema por ele.
Como resolver: Prefira devolver a decisão ao time com facilitação, perguntas e contexto (propósito, dados). Descarte o líder decidir sozinho, salvo emergência ou segurança em jogo.

📝 **Como cai:** Problema recorrente com várias soluções propostas e nenhum diagnóstico.
Como resolver: Procure a alternativa que diagnostica antes de escolher solução: busca a causa-raiz (5 porquês, Ishikawa, A3), verifica primeiro a hipótese mais provável e só encerra depois de medir o indicador. Se o problema já afeta clientes, contenha primeiro e investigue logo depois.

📝 **Como cai:** Pergunta conceitual sobre modelo com nome (Pink, Herzberg, Lencioni, Johari, escuta ativa, GROW).
Como resolver: Lembre o conteúdo exato do modelo e descarte distratores de modelos vizinhos (Maslow no lugar de Pink, SMART no lugar de GROW, escuta seletiva no lugar dos níveis).

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
Influência por admiração, sem cargo nem expertise	**Poder de referência**
Atender plenamente às preocupações das duas partes	**Colaborar (Thomas-Kilmann)**
Prazo curto e metas de importância moderada	**Conciliar (meio-termo)**
Ganhar virou mais importante que resolver; ataques pessoais	**Speed Leas nível 3: facilitação estruturada**
Intenção de destruir o outro lado	**Nível 5 (guerra): separar as partes e intervenção externa**
Objetivo, realidade, opções e ação	**GROW**
Base da pirâmide de disfunções do time	**Ausência de confiança (Lencioni)**
Autonomia, maestria e propósito	**Daniel Pink**
Problema, causa, contramedidas e plano em uma folha	**Relatório A3**
Ir ver o trabalho onde ele acontece	**Gemba walk**
Sala que torna visível toda a informação de um programa grande	**Obeya**
Seguir, adaptar e criar	**Shu-Ha-Ri**
Temas propostos e votados na hora, conversa com tempo limitado	**Lean coffee**
Alto poder e baixo interesse	**Manter satisfeito, informando nos marcos principais**

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

- Líder define o quê e o porquê; o time decide o como.
- Não sabe: treine. Sabe e precisa evoluir: coaching. Precisa da experiência de alguém: mentoria.
- A intervenção no conflito é proporcional ao nível de Speed Leas.
- Salário é higiênico; autonomia, maestria e propósito motivam.
- Antes de escalar, estimule o time a falar direto com quem está envolvido e leve dados e opções ao nível certo.

Essenciais para a prova (checklist)

- ☐ Sei diferenciar treinamento, coaching, mentoria e consultoria e quando usar cada um.
- ☐ Sei aplicar os quatro estilos da liderança situacional a perfis de competência e motivação.
- ☐ Sei identificar os cinco níveis de conflito de Speed Leas e a intervenção de cada um.
- ☐ Sei quando cada modo de Thomas-Kilmann é a escolha adequada.
- ☐ Sei as cinco disfunções de Lencioni na ordem.
- ☐ Sei os modelos de motivação de Pink e Herzberg e os efeitos de recompensas mal desenhadas.
- ☐ Sei reconhecer poder de referência, especialista, legítimo, recompensa e coercitivo.
- ☐ Sei usar 5 porquês, Ishikawa e A3 e quando conter antes de investigar.
- ☐ Sei como funciona uma comunidade de prática e outras formas de espalhar conhecimento.
- ☐ Sei calcular canais de comunicação com n(n-1)/2.

Questões de revisão do capítulo

R2.1 · Domínio II. Liderança

Uma diretora quer entender por que os times atrasam, mas conhece a situação só por relatórios. Um coach sugere que ela vá até onde o trabalho acontece, observe e converse com as pessoas. Qual prática Lean é essa?

A. Gemba walk
B. Poka-yoke
C. Kaizen blitz
D. Heijunka

R2.2 · Domínio II. Liderança

Uma startup de saúde contrata desenvolvedores com base em um teste técnico de quatro horas. O último contratado tem excelente domínio técnico, mas se recusa a programar em par, rejeita revisões e trabalha isolado, o que vem gerando atritos. O time vai contratar mais duas pessoas e participa do processo seletivo. O que o praticante deve sugerir?

A. Aumentar a dificuldade do teste técnico para selecionar os candidatos mais experientes
B. Contratar por período de experiência e decidir com base na entrega individual de cada um
C. Incluir perguntas comportamentais sobre situações reais de colaboração e ajuda a colegas
D. Exigir certificação em algum framework de trabalho iterativo como pré-requisito da vaga

R2.3 · Domínio II. Liderança

Em uma operadora de telecom, o time de identidade mantém a biblioteca de autenticação usada por nove aplicativos. O time do aplicativo de autoatendimento precisa de um novo método de login, mas o backlog dos donos da biblioteca está ocupado pelas próximas seis semanas. Os desenvolvedores do autoatendimento conhecem bem o código da biblioteca. Qual é a MELHOR abordagem?

A. Copiar a biblioteca para o repositório do autoatendimento e manter ali uma versão própria do código
B. Permitir que o time do autoatendimento contribua com a mudança, revisada pelo time de identidade
C. Pedir ao gerente das duas áreas que reordene o backlog do time de identidade
D. Implementar o novo login dentro do aplicativo, contornando a biblioteca até ela evoluir

R2.4 · Domínio II. Liderança

Uma operadora de telecom fundiu duas áreas: um time ganhou um novo gerente e perdeu dois colegas transferidos. Na primeira reunião diária após o anúncio, as pessoas falam pouco, respondem com monossílabos e evitam olhar para a câmera. A Scrum Master tinha planejado iniciar hoje uma nova dinâmica de refinamento. O que ela deve fazer PRIMEIRO?

A. Pedir ao novo gerente que explique os motivos da reorganização em uma reunião aberta com toda a área de operações
B. Reconhecer abertamente o impacto da mudança e abrir espaço para o time falar do que está sentindo
C. Seguir com a nova dinâmica de refinamento para manter o time focado no trabalho da iteração
D. Adiar a dinâmica e cancelar as reuniões da semana para dar ao grupo tempo de adaptação

Respostas comentadas

R2.1. A) Gemba walk

Por quê: Gemba é o lugar onde o valor é criado. Na gemba walk, o líder vai até o trabalho real, observa e pergunta com respeito, em vez de depender de relatórios. Kaizen blitz é um evento curto e intenso de melhoria, poka-yoke é um mecanismo à prova de erros e heijunka é o nivelamento da produção.

✗ B) Incorreta. Mecanismo à prova de erros.

✗ C) Incorreta. Evento curto e intenso de melhoria.

✗ D) Incorreta. Nivelamento da produção.

> **Dica de prova:** Ir ver o trabalho real: gemba walk.

R2.2. C) Incluir perguntas comportamentais sobre situações reais de colaboração e ajuda a colegas

Por quê: Times colaborativos dependem de pessoas dispostas a compartilhar trabalho, conhecimento e responsabilidade. Na seleção, entrevistas comportamentais, que exploram situações vividas pelo candidato, complementam a avaliação técnica e permitem perceber valores como abertura a feedback e ajuda mútua.

✗ A) Incorreta. O problema da última contratação não foi falta de habilidade técnica; um teste mais difícil não revela o comportamento colaborativo.

✗ B) Incorreta. Avaliar a pessoa pela entrega individual reforça justamente o trabalho isolado que causou o atrito.

✗ D) Incorreta. Uma certificação comprova conhecimento de conteúdo, não a disposição para colaborar no dia a dia.

> **Dica de prova:** Contratação para time colaborativo: avalie comportamento com exemplos reais, não só habilidade técnica.

R2.3. B) Permitir que o time do autoatendimento contribua com a mudança, revisada pelo time de identidade

Por quê: Inner source aplica práticas de código aberto dentro da empresa: componentes compartilhados aceitam contribuições de outros times, com revisão pelos donos. Isso aproveita o conhecimento distribuído na organização, reduz filas entre times e evita cópias divergentes do mesmo ativo.

✗ A) Incorreta. Uma cópia própria cria duas versões do componente de segurança, que passam a divergir e dobram o esforço de manutenção e de correção de falhas.

✗ C) Incorreta. Escalar ao gerente antes de tentar uma solução entre os times desloca a decisão e só troca a espera de um item por outro no backlog do time de identidade.

✗ D) Incorreta. Contornar o componente compartilhado cria dívida técnica justamente na autenticação, área em que a duplicação aumenta o risco.

> **Dica de prova:** Componente compartilhado com fila longa: o time que precisa contribui, o time dono revisa.

R2.4. B) Reconhecer abertamente o impacto da mudança e abrir espaço para o time falar do que está sentindo

Por quê: Consciência social é a competência de inteligência emocional de perceber as emoções dos outros e o clima do grupo. Depois de uma reorganização, sinais como silêncio e respostas curtas indicam tensão. O líder reconhece os sentimentos e dá espaço para o time falar antes de introduzir novas mudanças.

✗ A) Incorreta. Explicar os motivos pode ajudar mais tarde, mas não substitui reconhecer o que o time sente agora.

✗ C) Incorreta. Ignora os sinais claros de tensão; o time dificilmente vai se engajar em uma dinâmica nova nesse estado.

✗ D) Incorreta. Cancelar as reuniões reduz o contato justamente quando o time precisa de espaço para conversar.

Dica de prova: Time abalado por mudança: reconheça e acolha os sentimentos antes de introduzir algo novo.

03

CAPÍTULO 3 · 19% DA PROVA

Domínio III. Produto

Aqui eu preciso dominar o backlog e o valor: refinar e priorizar com os stakeholders, dividir histórias, estimar em conjunto, gerenciar incrementos, visualizar o trabalho e medir se o valor esperado chegou. É o menor peso (19%), mas concentra cálculos (EMV, ROI, VPL, TIR, payback, WSJF, pontuação ponderada) e muitas técnicas com nome (MoSCoW, Kano, story mapping, mapa de impacto, Buy a feature, Remember the future). Os cenários pedem a ação que mantém o backlog como lista única ordenada por valor, entrega fatias verticais de ponta a ponta e mede resultado, não volume de funcionalidades ou pontos. Onde mais erro: confundir Definição de Pronto com Definição de Preparado, MVP com funcionalidade mínima comercializável e entrega de funcionalidade com entrega de valor.

Neste capítulo

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

Tópicos-chave

Backlog DEEP e ordenação

Backlog bom é detalhado adequadamente, estimado, emergente e priorizado: itens distantes ficam grandes e grosseiros e se dividem quando se aproximam do topo. É uma lista única ordenada, sem empates entre itens. Ordenar o Product Backlog é responsabilidade do Product Owner, que pode delegar sem deixar de responder por isso. Backlog inchado pede revisões periódicas com os solicitantes para confirmar, ajustar ou retirar itens.

Histórias de usuário (3 Cs e INVEST)

Modelo de Ron Jeffries: cartão, conversa e confirmação. História boa é negociável: descreve a necessidade e o porquê, e os detalhes surgem da conversa entre PO e time; história sem o 'para que' não tem valor declarado. Granularidade do maior ao menor: tema, épico, história de usuário e tarefa.

Fatiamento de histórias

Divida na vertical: cada fatia atravessa as camadas e entrega valor (banco, back-end e interface separados não entregam nada). Padrões cobrados: caminho feliz primeiro e uma história por exceção; uma história por variação de regra, na ordem de valor; função com tela simples primeiro e interface sofisticada depois; fazer funcionar e depois fazer rápido; fluxo fino de ponta a ponta antes das etapas do meio.

Critérios de aceitação, BDD e Definições

Critério testável tem métrica, limite e condição de medição; adjetivos como 'rápido' não são testáveis. Formato Dado/Quando/Então indica BDD. Definição de Preparado controla a entrada do item na sprint e deve ser curta (guia, não portão); Definição de Pronto controla a saída. O que é comum a todos os itens vai para a Definição de Pronto; o específico fica nos critérios de aceitação.

Estimativa relativa

Pontos medem tamanho relativo, não horas: amarrá-los a dinheiro ou tempo distorce as estimativas. Escala instável se corrige com uma história de referência. Quem faz o trabalho estima; a PO esclarece e negocia escopo. Não reestime histórias prontas pelo tempo gasto, porque a velocidade já absorve a diferença.

Técnicas de estimativa

Muitos itens e pouco tempo: estimativa por afinidade, agrupando por tamanho. Épicos de roadmap pouco detalhados: tamanhos de camiseta (P, M, G, GG). Carta infinita ou valores enormes no planning poker indicam épico, que deve ser dividido no refinamento antes da planning.

Técnicas de priorização

MoSCoW: must, should, could e won't have this time (W não entra nesta release, mas pode voltar); com quase tudo Must nada é priorizado, limite os Must a cerca de 60% do esforço. Método dos 100 pontos, Buy a feature (dinheiro fictício e negociação) e dot voting (rápido e visual em grupo). Ponderação relativa soma benefício e penalidade e divide por custo e risco. Pareto: poucas causas, muitos efeitos.

Custo do atraso e WSJF

Custo do atraso é o valor perdido a cada período de espera. WSJF = (valor para usuário e negócio + criticidade no tempo + redução de risco ou habilitação de oportunidade) / tamanho; o maior custo do atraso nem sempre vence. Valor não é só receita: redução de risco e habilitação também contam. Habilitadores ficam visíveis no mesmo backlog e disputam capacidade por critério econômico.

Modelo de Kano

Básico: a ausência causa forte insatisfação e a presença é o mínimo esperado. Desempenho: satisfação cresce na proporção da quantidade ou qualidade. Encantamento: surpreende e não gera insatisfação se faltar. Indiferente não afeta a satisfação. Com o tempo, o encantamento de ontem vira o básico de hoje.

Métricas financeiras

ROI = (benefício - custo) / custo, não benefício / custo nem ganho absoluto. VPL maior indica mais valor, e o prazo do retorno já está no desconto. TIR: escolha a maior entre opções comparáveis, desde que acima da taxa mínima de atratividade. Payback = investimento / benefício por período. Custo de oportunidade é o valor da melhor alternativa que você deixa de fazer.

MVP x funcionalidade mínima comercializável (MMF)

MVP serve para aprender e testar uma hipótese, rapidamente; MMF é o menor conjunto que já tem valor reconhecido para ser vendido ou usado. Valide a demanda com o MVP antes de construir o release mínimo comercializável. Release bom se organiza em MMFs liberadas quando completas, não em histórias soltas nem componentes.

Visão, roadmap e descoberta

Visão (elevator pitch) resume para quem é o produto, o problema e o diferencial. Roadmap agora-próximo-depois descreve resultados buscados, com detalhe decrescente com o horizonte; datas longas se comunicam como faixas com nível de confiança. OKR: objetivo qualitativo e resultados-chave mensuráveis. Hipótese testável: acreditamos que X gerará Y; saberemos quando Z.

Story mapping, mapa de impacto e árvore de oportunidades

Story mapping organiza as histórias pela jornada do usuário; a primeira release pega a versão mais simples de cada atividade, de ponta a ponta, e não uma atividade completa. Mapa de impacto: meta, atores, impactos (mudança de comportamento) e entregas (funcionalidades). Árvore de oportunidades: resultado, oportunidades, soluções e experimentos. Jobs to be done pergunta o progresso que o cliente quer fazer.

Jogos colaborativos e personas

Remember the future: imaginar o produto já lançado e descrever, no passado, o que ele fez. Prune the product tree: tronco é o produto, galhos são áreas, folhas são ideias posicionadas pelo momento. Personas representam necessidades, objetivos e comportamentos; a persona primária orienta o que vem primeiro e as secundárias não podem prejudicá-la.

Incremento, Review e liberação

Item fora da Definição de Pronto não entra no Incremento nem é apresentado na Review: volta ao Product Backlog, e a iteração não se estende. Liberável é obrigação do time a cada iteração; liberado é decisão de negócio da PO. Review não é cerimônia de aceite: o feedback alimenta o backlog. Pronto vale por item; critérios de liberação (carga, suporte, reversão) valem para o conjunto.

Visualização do trabalho

Limite de WIP escrito no topo de cada coluna; colunas divididas em 'fazendo' e 'feito' revelam espera escondida. Raias mostram classe de serviço, marcadores de bloqueio mostram trabalho parado, cores mostram tipo de trabalho (alocação de capacidade). Trabalho invisível, como pedidos por telefone ou descoberta, vira cartão ou trilha própria. Quadro confiável é atualizado por quem faz o trabalho; idade do item mostra o que está envelhecendo frente à SLE.

Medir valor entregue

Funcionalidade entregue não é valor entregue: meça indicadores de resultado (adoção, frequência de uso, CSAT contra a linha de base). Para julgar cedo, use indicadores antecedentes ligados ao resultado consequente; para isolar o efeito, compare coortes expostas e não expostas no mesmo período. Downloads são métrica de vaidade; NPS = promotores (9 e 10) menos detratores (0 a 6). Alguém do negócio precisa medir o benefício depois do lançamento.

Definir valor além da receita

Valor de conformidade é o prejuízo evitado (multas, sanções, parar a operação). Segurança: modelagem de ameaças no refinamento e testes automatizados no pipeline como parte da Definição de Pronto. Privacidade, acessibilidade e sustentabilidade entram como critérios de aceitação ou de sucesso desde o início, não como revisão no fim. EBM: valor atual, valor não realizado, tempo para o mercado e capacidade de inovar.

Macetes para memorizar

💡 MoSCoW

Must, Should, Could, Won't have THIS TIME: o W não é 'nunca', é 'não agora'.

💡 Ready x Done

Ready é a porta de ENTRADA da sprint; Done é a porta de SAÍDA para o Incremento.

💡 MVP x MMF

MVP serve para APRENDER (testar a hipótese, em semanas); MMF serve para VENDER ou USAR (menor conjunto com valor reconhecido). Primeiro valide com o MVP, depois construa o release comercializável.

💡 DEEP

Detalhado adequadamente, Estimado, Emergente, Priorizado: backlog 'profundo' é detalhado no topo e raso no fundo.

💡 Fórmulas financeiras

EMV = P x I. ROI = (benefício - custo) / custo. Payback = investimento / benefício por mês. VPL e TIR: maior vence (TIR só acima da taxa mínima).

💡 Kano

Básico irrita se falta; Desempenho é 'quanto mais, melhor'; Encantamento surpreende e envelhece até virar básico.

💡 Granularidade

Tema > Épico > História > Tarefa: 'TEHT', do maior para o menor.

Tabela de memorização: não confunda

Quando é...	...e quando é
Definição de Preparado (Ready) Entrada da sprint: condições mínimas para o item ser selecionado; longa demais vira portão burocrático.	**Definição de Pronto (Done)** Saída: qualidade exigida para o item entrar no Incremento; o que é comum a todos os itens fica aqui.

MVP Objetivo é aprender e testar a demanda em semanas.	**Funcionalidade mínima comercializável (MMF)** Objetivo é gerar valor vendável ou utilizável, sem foco em testar hipótese.
Critérios de aceitação Regras específicas daquele item.	**Definição de Pronto** Itens comuns a todos (código revisado, testes automatizados, nenhuma vulnerabilidade crítica).
Kano básico A falta irrita e a presença é o mínimo esperado.	**Kano encantamento** A presença surpreende e a falta não gera insatisfação; tende a virar básico com o tempo.
ROI Retorno relativo: (benefício - custo) / custo, para comparar opções de investimentos diferentes.	**Payback** Tempo para recuperar o investimento: investimento / benefício por período.
Liberável Obrigação do time a cada iteração: incremento em condição de ir para produção.	**Liberado** Decisão de negócio da PO sobre quando expor ao cliente.
Tamanho de camiseta Épicos de roadmap ainda com uma frase, comparação grosseira rápida.	**Estimativa por afinidade** Backlog novo com muitos itens (como 200) que precisa de estimativa relativa inicial em poucas horas.

Números que caem

Item	Valor
Peso do domínio na prova	**19%**
Valor monetário esperado (EMV)	**Probabilidade x impacto (20% x 50 mil = 10 mil)**
ROI	**(benefício - custo) / custo**
Payback	**Investimento / benefício por período (200 mil / 50 mil por mês = 4 meses)**
WSJF	**(valor + criticidade no tempo + redução de risco ou habilitação) / tamanho**
NPS	**% de promotores (notas 9 e 10) menos % de detratores (notas 0 a 6)**
Must no MoSCoW	**Cerca de 60% do esforço**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Demonstrar na Review os itens com testes pendentes, sinalizando o que falta.
Como não cair: Item fora da Definição de Pronto não é apresentado: dá falsa ideia de progresso. Mostre só o pronto e devolva o resto ao backlog.

⚠ **Alerta de prova:** Estender a iteração alguns dias para atingir a meta inteira.
Como não cair: A duração é fixa; estender quebra a cadência e adia a inspeção e a adaptação.

⚠ **Alerta de prova:** Dividir uma história grande em banco de dados, back-end e interface.
Como não cair: Fatias horizontais não entregam valor sozinhas: fatie na vertical, de ponta a ponta.

⚠ **Alerta de prova:** Fixar uma conversão de pontos em horas para faturar ou cobrar.
Como não cair: Pontos são tamanho relativo; amarrá-los a tempo ou dinheiro distorce as estimativas.

⚠ **Alerta de prova:** Reestimar uma história pronta porque ela levou o dobro do tempo.
Como não cair: Mantenha a estimativa original e use o aprendizado em itens parecidos; a velocidade já absorve a diferença.

⚠ **Alerta de prova:** Calcular ROI como benefício dividido por custo ou escolher pelo maior ganho absoluto.
Como não cair: ROI = (benefício - custo) / custo; uma opção menor pode ter ROI maior.

⚠ **Alerta de prova:** Escolher o épico que gera receita mais cedo quando o critério pedido é VPL.
Como não cair: O VPL já desconta o tempo: vence o maior VPL.

⚠ **Alerta de prova:** No WSJF, escolher a funcionalidade com o maior custo do atraso.
Como não cair: WSJF divide o custo do atraso pelo tamanho; uma funcionalidade menor pode ter WSJF maior.

⚠ **Alerta de prova:** Calcular o EMV ignorando a probabilidade (usar o impacto inteiro).
Como não cair: EMV = probabilidade x impacto: 20% de 50 mil = 10 mil.

⚠ **Alerta de prova:** Detalhar agora em histórias um épico que só começa daqui a meses.
Como não cair: Item distante fica grande e grosseiro; detalhe quando ele se aproximar do topo do backlog.

⚠ **Alerta de prova:** Copiar em cada história os mesmos itens de qualidade, ou deixar requisitos não funcionais como histórias soltas no fim da fila.
Como não cair: Itens comuns e requisitos transversais (acessibilidade, auditoria) viram restrições na Definição de Pronto ou critérios das histórias afetadas.

⚠ **Alerta de prova:** Comemorar o volume de pontos entregues ou os downloads como prova de valor.
Como não cair: Pontos medem esforço estimado e downloads são vaidade; valor se mostra com indicadores de resultado como retenção, adoção e frequência.

⚠ **Alerta de prova:** Na primeira linha do story map, completar uma atividade inteira antes das outras.
Como não cair: A primeira release pega o mínimo de cada atividade, formando um fluxo de ponta a ponta que já possa ser usado.

⚠ **Alerta de prova:** No mapa de impacto, colocar funcionalidades no nível de impactos.
Como não cair: Impactos são mudanças de comportamento dos atores; funcionalidades ficam no nível de entregas.

Como cai na prova

📝 **Como cai:** Cálculo curto (EMV, ROI, payback, WSJF, pontuação ponderada, VPL, TIR) com alternativas que usam a conta errada.
Como resolver: Escreva a fórmula antes de olhar as alternativas, faça a conta e confira se a alternativa certa traz a interpretação correta. Distratores usam a razão invertida, ignoram a probabilidade ou somam sem aplicar pesos.

📝 **Como cai:** Fim de iteração com itens incompletos e pressão para mostrar tudo ou estender.
Como resolver: Só entra no Incremento e na Review o que atende à Definição de Pronto; o resto volta ao Product Backlog; a duração não muda.

📝 **Como cai:** História grande, ambígua ou que cresce a cada 'e se'; pede a melhor forma de dividir.
Como resolver: Procure a alternativa com fatia vertical que entrega valor: caminho feliz primeiro, uma história por regra ou exceção, interface simples primeiro, desempenho como história separada.

📝 **Como cai:** Stakeholders disputando prioridade ou backlog sem ordem clara.
Como resolver: Prefira lista única ordenada por critério de valor transparente (custo do atraso, WSJF, pontuação acordada) com clientes reais na priorização; descarte votação de quem grita mais ou classificar tudo como alta.

📝 **Como cai:** Funcionalidade entregue no prazo, e a diretoria quer saber se deu certo.
Como resolver: Escolha indicador de resultado ligado à meta (adoção, frequência, CSAT contra a linha de base, coortes), não volume de funcionalidades, pontos ou cumprimento de prazo.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
Dado / Quando / Então	**BDD**
Cartão, conversa e confirmação	**3 Cs de Ron Jeffries**
Valor perdido a cada semana de espera	**Custo do atraso (base do WSJF)**
Dinheiro fictício para comprar funcionalidades	**Buy a feature**
Cada stakeholder distribui 100 pontos	**Método dos 100 pontos**
Imaginar o produto já lançado e falar no passado	**Remember the future**
Árvore com tronco, galhos e folhas posicionadas por tempo	**Prune the product tree**
Resultado, oportunidades, soluções e experimentos	**Árvore de oportunidades e soluções**
Meta, atores, impactos e entregas	**Mapa de impacto**
Objetivo qualitativo com KR1, KR2	**OKR**
Carta infinita ou valores enormes no planning poker	**Épico: dividir no refinamento antes da planning**
Quase tudo classificado como Must	**Limitar os Must a cerca de 60% do esforço**
Benefício + penalidade divididos por custo e risco	**Ponderação relativa**
80% das reclamações vêm de 20% das funcionalidades	**Princípio de Pareto**

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

- Item fora da Definição de Pronto volta ao backlog: não é demonstrado e a iteração não se estende.
- Fatie na vertical; cada fatia entrega valor de ponta a ponta.
- Pontos são tamanho relativo, estimados por quem faz, e não horas nem meta.
- Funcionalidade entregue não é valor entregue: meça o resultado depois do lançamento.
- Valor inclui conformidade, segurança, privacidade e redução de risco, e não só receita.

Essenciais para a prova (checklist)

- ☐ Sei calcular EMV, ROI, payback e WSJF e escolher entre opções por VPL e TIR.
- ☐ Sei aplicar MoSCoW, Kano, 100 pontos, Buy a feature, dot voting e ponderação relativa.
- ☐ Sei dividir histórias na vertical com os principais padrões de fatiamento.
- ☐ Sei diferenciar Definição de Preparado, Definição de Pronto e critérios de aceitação.
- ☐ Sei diferenciar MVP e funcionalidade mínima comercializável.

- ☐ Sei usar story mapping, mapa de impacto, árvore de oportunidades, personas e OKR.
- ☐ Sei quando usar afinidade, tamanho de camiseta e planning poker.
- ☐ Sei montar um quadro com limites de WIP, colunas fazendo/feito, raias por classe de serviço e bloqueios.
- ☐ Sei escolher indicadores de resultado e antecedentes para medir o valor de um incremento.
- ☐ Sei explicar a diferença entre liberável e liberado e o papel da Review.

Questões de revisão do capítulo

R3.1 · Domínio III. Produto

Um sistema de gestão escolar tem 140 funções. Os dados de uso mostram que 30 delas foram acessadas por menos de 1% dos clientes no último ano, mas cada liberação exige testar todas, e essas 30 dependem de componentes antigos que impedem a migração para uma nova plataforma. A Product Owner precisa decidir o que fazer com elas. Qual é a MELHOR decisão?

A. Descontinuar as funções de pouco uso, avisando os clientes afetados e indicando alternativas
B. Migrar as 30 funções para a nova plataforma, preservando a compatibilidade com as versões antigas
C. Manter as funções e automatizar os testes de regressão de cada uma delas antes da migração
D. Esconder as funções do menu principal, mantendo o código para quem ainda souber acessá-las

R3.2 · Domínio III. Produto

A área de vendas de uma empresa de software de gestão para restaurantes exige datas exatas para os oito itens do roadmap dos próximos doze meses, para prometê-las aos clientes. O time tem velocidade estável, mas os itens do segundo semestre ainda são épicos pouco detalhados. Como o praticante deve orientar a comunicação do roadmap?

A. Suspender a divulgação de datas aos vendedores até que os itens entrem em desenvolvimento
B. Compartilhar datas mais firmes para o próximo trimestre e janelas mais amplas, com nível de confiança, para o resto
C. Repassar as datas estimadas pelo time sem ressalvas, para não gerar insegurança nos clientes
D. Fixar uma data para cada um dos oito itens e reservar uma margem de segurança de 20% em cada uma

R3.3 · Domínio III. Produto

O comitê de produto de uma seguradora tem orçamento para uma de duas iniciativas, ambas com investimento de R$ 1 milhão e horizonte de cinco anos. A iniciativa A tem TIR de 18% ao ano. A iniciativa B tem TIR de 9% ao ano, mas um diretor destaca que ela gera o maior retorno nominal somado no período. A taxa mínima de atratividade da empresa é de 12% ao ano. Qual iniciativa o comitê deve escolher?

A. As duas em sequência, já que ambas têm TIR positiva e se pagam dentro do horizonte
B. A iniciativa B, por gerar o maior retorno nominal acumulado nos cinco anos
C. A iniciativa B, porque uma TIR menor indica um risco menor para o investimento da empresa
D. A iniciativa A, cuja TIR de 18% supera a taxa mínima de 12% exigida pela empresa

R3.4 · Domínio III. Produto

Os dados de suporte mostram que cerca de 80% das reclamações vêm de 20% das funcionalidades. Qual princípio o Product Owner está aplicando ao usar isso para priorizar melhorias?

A. Lei de Hofstadter
B. Lei de Brooks
C. Princípio de Pareto
D. Efeito Dunning-Kruger

Respostas comentadas

R3.1. A) Descontinuar as funções de pouco uso, avisando os clientes afetados e indicando alternativas

Por quê: Otimizar os incrementos de valor, como pede o ECO, inclui olhar para o custo total de manter o que já existe. Funcionalidades que quase ninguém usa são uma forma de desperdício: consomem testes, manutenção e atenção do time. Descontinuá-las de forma planejada, com comunicação e alternativas para os afetados, reduz o custo e libera capacidade para o que gera valor.

✗ B) Incorreta. Migrar as funções gasta esforço considerável em algo que quase ninguém usa, além de levar a complexidade para a nova plataforma.

✗ C) Incorreta. Automatizar testes reduz parte do esforço, mas mantém o custo de manutenção e o bloqueio causado pelos componentes antigos.

✗ D) Incorreta. Esconder do menu mantém todo o custo de manutenção e cria uma situação confusa para os poucos usuários.

> **Dica de prova:** Funcionalidade sem uso também custa: descontinuar com comunicação pode gerar mais valor que mantê-la.

R3.2. B) Compartilhar datas mais firmes para o próximo trimestre e janelas mais amplas, com nível de confiança, para o resto

Por quê: Previsões de longo prazo carregam mais incerteza, como mostra o cone da incerteza. Um roadmap comunicado com níveis de confiança separa o que é quase certo (próximas iterações) do que é provável ou possível (horizontes distantes), e as janelas se estreitam à medida que os itens são refinados e o time ganha dados. Isso preserva a confiança das áreas de negócio.

✗ A) Incorreta. Esconder as previsões prejudica o planejamento comercial; o problema está na forma de comunicar, não em comunicar.

✗ C) Incorreta. Repassar estimativas como compromissos ignora a incerteza e transforma previsões em promessas aos clientes.

✗ D) Incorreta. Uma margem fixa dá aparência de precisão a épicos ainda vagos, e as promessas tendem a ser quebradas.

> **Dica de prova:** Datas de roadmap com horizonte longo: comunique faixas com nível de confiança, mais firmes no curto prazo.

R3.3. D) A iniciativa A, cuja TIR de 18% supera a taxa mínima de 12% exigida pela empresa

Por quê: A taxa interna de retorno (TIR) é a taxa de desconto que zera o VPL dos fluxos de caixa de um investimento. Uma iniciativa é atraente quando a TIR supera a taxa mínima de atratividade (o custo de oportunidade do capital); entre opções de porte e prazo semelhantes, prefere-se a de maior TIR. Somar retornos nominais ignora o valor do dinheiro no tempo.

✗ A) Incorreta. Uma TIR positiva não basta: B fica abaixo da taxa mínima de 12% e, por esse critério, destruiria valor.

✗ B) Incorreta. Somar valores nominais ignora o valor do dinheiro no tempo; a TIR de 9% mostra que B não remunera o capital à taxa exigida.

✗ C) Incorreta. A TIR mede retorno, não risco; uma TIR menor não indica investimento mais seguro.

> **Dica de prova:** TIR: escolha a maior entre opções comparáveis, desde que acima da taxa mínima de atratividade.

R3.4. C) Princípio de Pareto

Por quê: O princípio de Pareto diz que uma pequena parte das causas costuma gerar a maior parte dos efeitos, algo como 80/20. Priorizar as poucas funcionalidades que concentram as reclamações gera o maior ganho com o menor esforço. O gráfico de Pareto ordena as causas por frequência para tornar isso visível.

✗ A) Incorreta. Diz que tudo demora mais do que o previsto.

✗ B) Incorreta. Trata de adicionar pessoas a projetos atrasados.

✗ D) Incorreta. Viés de autoavaliação, sem relação com priorização.

> **Dica de prova:** Poucas causas, muitos efeitos: priorize pelo princípio de Pareto.

04

CAPÍTULO 4 · 28% DA PROVA

Domínio IV. Entrega

Este domínio, empatado como o maior da prova (28%), cobra como eu entrego: feedback antecipado, métricas ágeis, impedimentos e riscos, desperdícios, melhoria contínua, envolvimento do cliente e otimização do fluxo. Caem muitos gráficos e contas (Lei de Little, eficiência de fluxo, CFD, burnup e burndown, AgEVM com CPI e SPI, Monte Carlo, percentis da SLE, gráfico de controle) e práticas técnicas (TDD, ATDD, integração contínua, feature toggles, blue-green, canário). A resposta certa costuma limitar o WIP, reduzir o tamanho do lote, embutir qualidade em vez de inspecionar no fim e usar dados do próprio time. Onde mais erro: lead time x tempo de ciclo, média x percentil e conclusões de métricas que viram meta.

Neste capítulo

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

Tópicos-chave

Lead time, tempo de ciclo, throughput e idade do item

Lead time vai da solicitação à entrega; tempo de ciclo vai do início à conclusão. Throughput é quantos itens terminam por período. Tempo de ciclo só existe para itens concluídos e olha para trás; a idade do item mostra o que ainda está em andamento e envelhecendo. Cliente reclama de prazo e o ciclo é curto: a demora está na fila antes do início.

Lei de Little

Relaciona médias de um sistema estável: WIP = throughput x lead time, ou tempo de ciclo = WIP / throughput. Com throughput constante, mais WIP significa tempo de ciclo maior; lead time dobrado com throughput fixo significa WIP dobrado. No portfólio, menos épicos em andamento com o mesmo ritmo de conclusão significa épicos concluídos mais cedo.

SLE e previsão probabilística

A expectativa de nível de serviço tem duas partes, tempo e probabilidade (como 85% dos itens em até 9 dias), e vem do histórico de tempo de ciclo, contado a partir do início do item. Para prometer prazo use percentil, não média nem o maior valor já registrado. Data fixa com multa alta pede percentil mais alto. Tempo de ciclo muito variável se resolve dimensionando os itens para caberem na SLE.

Simulação de Monte Carlo

Usa o throughput histórico para responder 'quando?' em faixas de probabilidade. O percentil é probabilidade acumulada: 85% para uma data significa que em 85% das execuções o escopo inteiro terminou até ela, não que 85% do escopo termina nela nem que a chance é de cair exatamente naquele dia. Quanto maior a confiança, mais tarde a data.

Diagrama de fluxo cumulativo (CFD)

Distância vertical entre faixas é o WIP; horizontal é o tempo de ciclo aproximado; inclinação é o throughput. Faixas paralelas com largura constante indicam fluxo estável; faixa que se alarga é gargalo ou acúmulo. Só a faixa de entrada crescendo indica demanda acima da capacidade: atue na entrada, não no WIP.

Burndown e burnup

Burnup separa progresso de mudança de escopo: escopo subindo quase tão rápido quanto o concluído põe a data em risco. Burndown acima da linha ideal indica atraso ou escopo adicionado; burndown subindo no meio da iteração indica escopo adicionado ou reestimado; burndown reto com time ocupado indica muito trabalho começado e nada pronto.

AgEVM

EV = percentual de pontos concluídos x orçamento total (BAC). PV = percentual de sprints decorridas x BAC. CPI = EV / AC e SPI = EV / PV; abaixo de 1 é custo acima do previsto ou atraso. Distratores trazem a razão invertida ou os percentuais isolados.

Velocidade e métricas que viram meta

Velocidade serve ao planejamento do próprio time, nunca à comparação entre times. Lei de Goodhart: quando a medida vira meta, deixa de ser boa medida (velocidade sobe sem mais entregas porque as estimativas inflaram). Composição do time mudou, recalibre a velocidade. Previsão de release: divida o restante pela maior e pela menor velocidade recentes e comunique a faixa.

Gráfico de controle

Variação grande dentro dos limites é causa comum: mude o sistema (itens menores, limite de WIP), não explique ponto por ponto. Ponto fora do limite é causa especial: investigue aquele caso. Excluir dados ou estreitar limites artificialmente só maquia a variação.

Métricas DORA

Frequência de implantação, lead time de mudanças, taxa de falha de mudanças e tempo de restauração. Muitas falhas pedem lotes menores e testes automatizados; aprovações de comitê externo não reduzem a taxa de falha e aumentam lead time e lote. Recuperação lenta pede implantação e reversão automatizadas e chaves de funcionalidade.

Desperdícios Lean

Valor agregado é o que o cliente reconhece e pagaria; retrabalho, filas e redigitação não são. Desperdícios cobrados: superprodução, estoque (funcionalidade pronta e não liberada, especificação cedo demais), passagens de bastão, processamento extra, reaprendizado e talento não utilizado. Muri é sobrecarga (ocupação de 100%) e mura é irregularidade (picos e vales na chegada do trabalho). Atividade exigida por conformidade não se elimina: minimize e automatize.

Mapa do fluxo de valor e eficiência de fluxo

Siga o item e meça o tempo parado, não a ocupação das pessoas; as esperas costumam pesar mais que o trabalho. Eficiência = tempo de trabalho / lead time total (trabalho + espera). Eficiência baixa pede menos fila, não gente mais rápida. Mapa futuro é hipótese: implemente, mapeie de novo e meça.

Teoria das restrições e utilização

O gargalo define a vazão: explore, subordine e eleve a restrição. Não empurre mais trabalho para o gargalo; subordine as outras etapas ao ritmo dele e use a folga para ajudá-lo. Utilização perto de 100% faz as filas explodirem: folga de capacidade é condição para fluxo rápido.

Riscos e impedimentos

Já está acontecendo e bloqueando é impedimento; pode acontecer é risco. Exposição total = soma de probabilidade x impacto de cada risco, acompanhada a cada iteração no risk burndown; burndown de riscos estável enquanto as funcionalidades andam indica mitigação parada. Pre-mortem imagina o fracasso antes de começar. Resposta a risco vai para o backlog, com dono e gatilho, escolhida com o time.

ROAM no SAFe

Resolved: o risco deixou de ser preocupação. Owned: alguém assume o acompanhamento de um risco que não se resolve no evento. Accepted: nada mais pode ser feito e o risco é aceito. Mitigated: existe plano definido para reduzir probabilidade ou impacto.

Qualidade embutida e testes

Inspeção no fim encontra defeitos; qualidade embutida evita que sejam criados. TDD: vermelho, verde, refatorar. ATDD: critérios de aceite como testes antes do código. Muitos testes de interface lentos e instáveis: desça para testes de unidade e mantenha poucos de ponta a ponta. Meta de cobertura gera testes vazios: use cobertura como diagnóstico. Defeito que chegou ao cliente é defeito escapado.

Integração e entrega contínuas

Servidor de CI não é integração contínua: integre ao tronco mudanças pequenas pelo menos uma vez por dia (trunk-based development). Entrega contínua deixa pronto para produção; implantação contínua vai para produção sozinha. Compile uma vez e promova o mesmo artefato por cada ambiente, mudando só a configuração. Feature toggles integram sem expor e separam implantação de liberação.

Liberação de baixo risco

Blue-green: dois ambientes e troca de tráfego com rollback rápido. Canário: libera a um subconjunto, monitora e só então amplia. Anéis vão do público mais tolerante e próximo (o próprio time) ao mais amplo e sensível (todos os clientes). Recurso novo e imaturo começa pelos adotantes iniciais.

Macetes para memorizar

💡 Lei de Little

W = T x L (WIP = Throughput x Lead time): 'WTL'. Mexeu num lado com T fixo, os outros dois andam juntos.

💡 Leitura do CFD

Vertical = Volume em andamento (WIP); Horizontal = Hora (tempo de ciclo); Inclinação = Intensidade da saída (throughput).

💡 AgEVM

Sempre EV em cima: CPI = EV/AC, SPI = EV/PV. Abaixo de 1 é ruim.

💡 Lead x ciclo

Lead time começa no Pedido (o Lead é do cliente); ciclo começa quando o time Inicia.

💡 Teoria das restrições

ESE: Explorar, Subordinar, Elevar o gargalo, nessa ordem.

💡 Muda, mura, muri

mURA = irregUlaR (picos e vales); mURI = sobrecaRga, até 100%.

💡 ROAM

Resolvido, dOno (Owned), Aceito, Mitigado: Owned = tem dono; Mitigated = tem plano.

Tabela de memorização: não confunda

Quando é...	...e quando é
Lead time Da solicitação (registro do pedido) à entrega: é o que o cliente sente.	**Tempo de ciclo** Do início do trabalho à conclusão: mede o trecho dentro do time e é a base da SLE.
Média Fica perto do meio da distribuição e não informa a chance de cumprir o prazo.	**Percentil** Associa prazo a probabilidade (85% em até X dias), tornando o risco explícito; nenhum percentil garante prazo.

Entrega contínua Todo incremento fica pronto para produção, e a decisão de implantar é humana.	**Implantação contínua** Cada mudança aprovada no pipeline vai para produção sozinha.
Risco Pode acontecer: registre com dono e gatilho, planeje a resposta no backlog.	**Impedimento** Já está acontecendo e bloqueando: remova ou escale, priorizando o que ameaça a Meta da Sprint.
Causa comum Variação dentro dos limites de controle: mude o sistema.	**Causa especial** Ponto fora do limite de controle: investigue aquele caso.
Blue-green Dois ambientes completos e troca de tráfego com rollback rápido.	**Canário** Expor a mudança a um subconjunto de usuários, monitorar e ampliar aos poucos.
Muri Sobrecarga, como ocupação de 100%: limite o WIP e deixe folga.	**Mura** Irregularidade, como picos e vales na chegada do trabalho: nivele a entrada.

Números que caem

Item	Valor
Peso do domínio na prova	**28%**
Lei de Little	**WIP = throughput x lead time (tempo de ciclo = WIP / throughput)**
Eficiência de fluxo	**Tempo de trabalho / (trabalho + espera); 3 dias em 20 = 15%**
AgEVM	**EV = % pontos concluídos x BAC; CPI = EV / AC; SPI = % pontos concluídos / % sprints decorridas**
Integração contínua	**Integrar ao tronco pelo menos uma vez por dia**
Métricas DORA	**4: frequência de implantação, lead time, taxa de falha e tempo de restauração**

Alertas de prova (pegadinhas)

⚠ **Alerta de prova:** Calcular CPI como custo real dividido pelo orçamento total, ou usar a razão invertida.
Como não cair: CPI = EV / AC, com EV = percentual de pontos concluídos x BAC; 375/800 é só o orçamento consumido e 1,25 é AC/EV.

⚠ **Alerta de prova:** No SPI, responder apenas o percentual concluído ou o percentual de sprints decorridas.
Como não cair: SPI = EV / PV = percentual de pontos concluídos / percentual de sprints decorridas (30% / 40% = 0,75).

⚠ **Alerta de prova:** Dividir o tempo de trabalho pelo tempo de espera para achar a eficiência do ciclo.
Como não cair: O denominador é sempre o lead time total (trabalho + espera): 9 / 32, cerca de 28%.

⚠ **Alerta de prova:** Com eficiência de fluxo baixa, concluir que as pessoas estão ociosas e devem pegar mais itens.
Como não cair: Mais itens ao mesmo tempo aumentam WIP e filas; baixa eficiência pede reduzir espera, não acelerar pessoas nem contratar.

⚠ **Alerta de prova:** Achar que throughput estável significa WIP estável.
Como não cair: Pela Lei de Little, lead time maior com throughput fixo só ocorre se o WIP cresceu.

⚠ **Alerta de prova:** Ler o percentil 85 do Monte Carlo como a data mais provável ou como 85% do escopo pronto.
Como não cair: É probabilidade acumulada de terminar todo o escopo até a data; ainda há 15% de chance de atraso.

⚠ **Alerta de prova:** Comunicar a mediana do tempo de ciclo como prazo de cada item, ou o maior valor já registrado.
Como não cair: A mediana falha em metade dos casos e o máximo é inútil; comunique o percentil (ex.: 85% em até 9 dias) a partir do início do item.

⚠ **Alerta de prova:** Tratar item de data fixa como urgente agora, furando o limite de WIP.
Como não cair: Data fixa não é urgente: comece com folga calculada pelo histórico de tempo de ciclo, com percentil maior quanto maior o custo de perder a data.

⚠ **Alerta de prova:** Aumentar o limite de WIP no pico ou quando a coluna lota.
Como não cair: Limite atingido pede ajudar a terminar o que está parado; elevar WIP aumenta sobrecarga e lead time. Item bloqueado continua no WIP.

⚠ **Alerta de prova:** Incluir aprovação de comitê externo ou espaçar implantações para reduzir falhas.
Como não cair: Pela DORA, lotes menores, implantações frequentes e testes automatizados no pipeline reduzem falhas; aprovação externa aumenta lead time e lote.

⚠ **Alerta de prova:** Usar a velocidade de um time como meta para os demais ou comparar times por ela.
Como não cair: Velocidade é ferramenta de planejamento do próprio time; virando meta, as estimativas inflam (Goodhart). Avalie melhoria contra a linha de base do próprio time.

⚠ **Alerta de prova:** Investigar cada item demorado dentro dos limites do gráfico de controle.
Como não cair: Dentro dos limites é causa comum: mude o sistema. Investigar ponto a ponto desperdiça esforço e pode aumentar a variação.

⚠ **Alerta de prova:** Considerar um risco 'mitigado' porque alguém está em contato com o fornecedor.
Como não cair: Sem plano definido não é Mitigated; se uma pessoa assumiu acompanhar o risco após o evento, é Owned.

⚠ **Alerta de prova:** Adotar várias mudanças de processo de uma vez na retrospectiva e votar qual funcionou.
Como não cair: Melhoria é experimento: uma mudança por vez, com métrica e prazo, para saber o que causou cada efeito.

Como cai na prova

📝 **Como cai:** Conta de fluxo ou de AgEVM com quatro alternativas numéricas e uma interpretação.
Como resolver: Escreva a fórmula, identifique os dados (BAC, AC, pontos, sprints, trabalho, espera), calcule e confira a interpretação; descarte razões invertidas e percentuais isolados.

📝 **Como cai:** Gráfico descrito em texto (CFD, burnup, burndown, controle, dispersão) e pergunta o diagnóstico.
Como resolver: Traduza a forma para o significado: faixa alargando = gargalo; escopo subindo = risco de data; ponto fora do limite = causa especial; dispersão sem padrão = itens de tamanhos muito diferentes.

📝 **Como cai:** Pedido de prazo ou previsão para o negócio.
Como resolver: Use dados históricos do próprio time e comunique faixa com probabilidade (percentil, Monte Carlo, maior e menor velocidade), nunca média como promessa firme.

📝 **Como cai:** Problema de qualidade ou de implantação e propostas de mais controle.
Como resolver: Prefira lotes menores, automação no pipeline e qualidade embutida (TDD, testes de unidade, verificação automática); descarte comitês, documentação prévia e inspeção no fim.

📝 **Como cai:** Fluxo lento, filas e pessoas ocupadas.
Como resolver: Limite o WIP, termine antes de começar, encontre e subordine-se ao gargalo, reduza passagens e o tamanho do lote; descarte contratar ou aumentar o WIP.

Gatilhos do enunciado

Se o enunciado trouxer...	Pense em
Faixa do CFD que se alarga	**Gargalo / acúmulo nessa etapa**
Só a faixa de entrada do CFD crescendo	**Demanda acima da capacidade: atuar na entrada**
Burnup com escopo subindo quase tão rápido quanto o concluído	**Data em risco: negociar escopo**
Velocidade subindo sem mais entregas	**Estimativas infladas, Lei de Goodhart**
Rodadas anônimas de estimativa com discussão até convergir	**Wideband Delphi**
Imaginar o fracasso antes de começar	**Pre-mortem**
Vermelho, verde, refatorar	**TDD**
Critérios de aceite escritos como testes antes do código	**ATDD**
Integrar sem expor ao usuário	**Feature toggles**
Condutor e navegador no mesmo código	**Programação em par**
Urgência que fura a fila	**Classe de serviço expedite (limitada e com critério de entrada)**
Problema que não pode esperar a retrospectiva	**Intraspectiva curta com o time**
Times de produto repetindo o mesmo trabalho de infraestrutura	**Plataforma de autoatendimento**
Quer arquitetura por domínios	**Organizar times por domínios (manobra inversa de Conway)**

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

- Limitar o WIP é a alavanca mais direta para reduzir o tempo de ciclo: pare de começar, comece a terminar.
- Previsão se comunica em faixa e probabilidade, nunca média como compromisso.
- Métrica que vira meta deixa de medir: velocidade não compara times.
- Qualidade se embute com lotes pequenos e automação, não com aprovação no fim.
- Ação de melhoria e resposta a risco só acontecem se entrarem no backlog com capacidade.

Essenciais para a prova (checklist)

- ☐ Sei diferenciar lead time, tempo de ciclo, throughput e idade do item.
- ☐ Sei aplicar a Lei de Little e calcular eficiência de fluxo.
- ☐ Sei ler CFD, burnup, burndown, gráfico de controle e gráfico de dispersão.
- ☐ Sei calcular EV, CPI e SPI no AgEVM.
- ☐ Sei interpretar percentis da SLE e de uma simulação de Monte Carlo.
- ☐ Sei as quatro métricas DORA e o que as melhora.
- ☐ Sei classificar desperdícios Lean, incluindo muda, mura e muri.
- ☐ Sei a diferença entre risco e impedimento e os estados do ROAM.
- ☐ Sei diferenciar TDD, ATDD, BDD, integração contínua, entrega contínua e implantação contínua.
- ☐ Sei quando usar feature toggles, blue-green, canário e anéis de liberação.

Questões de revisão do capítulo

R4.1 · Domínio IV. Entrega

Em um órgão público, os impedimentos de um time são citados na reunião diária, mas ninguém sabe dizer há quanto tempo cada um está aberto. Na retrospectiva, o time percebeu que dois bloqueios dependentes de outras secretarias ficaram parados por cinco semanas sem que ninguém de fora do time soubesse. Qual prática resolve MELHOR esse problema?

A. Pedir que cada desenvolvedor acompanhe os próprios impedimentos até que eles sejam resolvidos
B. Dedicar parte de cada retrospectiva à revisão dos impedimentos que continuam abertos no time
C. Abrir um chamado na central de serviços para cada impedimento relatado durante a reunião diária
D. Manter um quadro visível de impedimentos com a data de abertura e um prazo para escalar cada um

R4.2 · Domínio IV. Entrega

O time de faturamento de uma operadora de telecom desenvolve durante dez semanas as funcionalidades do trimestre e depois tem uma fase de testes de quatro semanas. Na última fase foram encontrados 180 defeitos, muitos difíceis de rastrear porque a causa estava em código escrito dois meses antes. Qual mudança resolve MELHOR o problema?

A. Iniciar a fase de testes duas semanas antes do término do desenvolvimento do trimestre
B. Ampliar a fase de testes do fim do trimestre de quatro para seis semanas de duração
C. Montar uma equipe dedicada a investigar a causa dos defeitos acumulados no trimestre
D. Testar cada história na mesma sprint em que ela é desenvolvida, em pequenos lotes

R4.3 · Domínio IV. Entrega

Uma empresa de software vai firmar contrato com uma companhia estadual de saneamento para desenvolver um sistema de leitura de hidrômetros, com entregas quinzenais. No contrato anterior com o mesmo cliente, os representantes dele faltavam às revisões, e os problemas só apareceram na entrega final. Qual cláusula é a MAIS importante incluir no novo contrato?

A. Transferir ao fornecedor as decisões de prioridade quando o cliente não comparecer
B. Prever a participação de representantes do cliente nas revisões e no refinamento do backlog
C. Fixar o escopo detalhado e as multas por atraso, protegendo as duas partes de mudanças no caminho
D. Prever uma aprovação formal do cliente ao final de cada fase, com prazo para responder

R4.4 · Domínio IV. Entrega

Ao mapear o fluxo de concessão de crédito imobiliário de um banco, o time precisa classificar as atividades do ponto de vista do cliente. Entre elas estão: avaliar a capacidade de pagamento do solicitante, redigitar os dados da proposta de um sistema em outro, corrigir erros encontrados no formulário e aguardar a assinatura do gerente regional. Qual delas agrega valor?

A. A redigitação dos dados entre sistemas, necessária para a proposta seguir no fluxo
B. A correção dos erros no formulário, que garante a qualidade da proposta enviada
C. A análise da capacidade de pagamento, que o cliente reconhece como parte do serviço
D. A espera pela assinatura do gerente, que protege juridicamente o banco e o cliente

Respostas comentadas

R4.1. D) Manter um quadro visível de impedimentos com a data de abertura e um prazo para escalar cada um

Por quê: Gestão visual torna o estado do trabalho, inclusive problemas, acessível a todos. Um quadro de impedimentos com data de abertura mostra a idade de cada bloqueio, e uma regra explícita de escalonamento, por exemplo após três dias, garante que impedimentos que dependem de outras áreas cheguem a quem tem autoridade para removê-los antes de afetar a entrega.

✗ A) Incorreta. Distribuir o acompanhamento individualmente esconde os bloqueios do restante do time e de quem poderia ajudar a removê-los.

✗ B) Incorreta. A retrospectiva ocorre ao fim da sprint; revisar os bloqueios só ali deixa que fiquem parados por semanas antes de alguém agir.

✗ C) Incorreta. Um chamado registra o pedido, mas fica fora da vista do time e não mostra a idade nem dispara o escalonamento quando o bloqueio passa do razoável.

> **Dica de prova:** Impedimento sem data e sem regra de escalonamento envelhece em silêncio: torne a idade visível.

R4.2. D) Testar cada história na mesma sprint em que ela é desenvolvida, em pequenos lotes

Por quê: Trabalhar em lotes grandes adia o feedback: quanto mais tempo passa entre criar um defeito e descobri-lo, mais caro é diagnosticá-lo e corrigi-lo. Lotes pequenos, testados assim que concluídos, encurtam esse ciclo, reduzem o risco de cada entrega e tornam o fluxo mais previsível.

✗ A) Incorreta. Antecipar a fase em duas semanas ainda testa grandes volumes muito tempo depois de o código ser escrito.

✗ B) Incorreta. Mais tempo de teste no fim acomoda o grande lote, mas mantém a demora entre escrever o código e descobrir o defeito.

✗ C) Incorreta. Investigar defeitos acumulados trata as consequências do lote grande, em vez de reduzir o tamanho do lote.

> **Dica de prova:** Lote grande com teste no fim gera defeitos caros: teste em lotes pequenos, logo após construir.

R4.3. B) Prever a participação de representantes do cliente nas revisões e no refinamento do backlog

Por quê: O Manifesto valoriza a colaboração com o cliente mais que a negociação de contratos. Isso não elimina o contrato: ele pode ser desenhado para favorecer a colaboração, por exemplo prevendo a participação do cliente em revisões e no refinamento. Quando o histórico mostra que a falta de envolvimento causou problemas, essa é a cláusula que mais protege o resultado.

✗ A) Incorreta. Decisões de prioridade pertencem ao lado do cliente; transferi-las ao fornecedor afasta ainda mais o cliente do produto.

✗ C) Incorreta. Escopo fixo e multas reforçam a negociação de contrato e desestimulam mudanças, sem resolver a falta de participação do cliente.

✗ D) Incorreta. Aprovações formais por fase são pontos de controle espaçados; não substituem a colaboração contínua durante o trabalho.

> **Dica de prova:** Contratos podem promover a colaboração: preveja a participação do cliente nos eventos de feedback.

R4.4. C) A análise da capacidade de pagamento, que o cliente reconhece como parte do serviço

Por quê: No Lean, valor é definido pelo cliente: uma atividade agrega valor se o cliente consideraria o produto menos valioso sem ela. Retrabalho e tempo em fila dificilmente são vistos como valor pelo cliente. Atividades que só existem pela forma atual do processo, como redigitar dados, adicionam custo sem valor e são candidatas à eliminação.

✗ A) Incorreta. Redigitar dados consome recursos sem que o cliente perceba qualquer ganho; é necessária só pela forma atual do processo.

✗ B) Incorreta. Corrigir erros é retrabalho; o cliente não pagaria a mais por uma correção que não deveria ser necessária.

✗ D) Incorreta. Tempo de espera nunca agrega valor para o cliente, ainda que a aprovação em si possa ser exigida pelo banco.

> **Dica de prova:** Valor agregado é o que o cliente reconhece e pagaria; retrabalho, filas e redigitação não são.

DIA DA PROVA

Estratégia de prova

A prova do PMI-ACP é situacional: a maioria dos enunciados descreve um time, uma organização e um problema concreto e pergunta a melhor ação, a próxima ação ou o que explica um dado. Quase sempre duas alternativas parecem razoáveis; a certa é a que segue os valores e princípios ágeis (colaborar, tornar visível, experimentar pequeno, devolver a decisão ao time, atacar a causa) e as erradas trazem controle, documento, aprovação, escalonamento precoce ou solução de sintoma. Também há itens conceituais diretos sobre modelos com nome e cálculos curtos de fluxo, valor e AgEVM. O exame vigente valoriza a agilidade empresarial: alinhar o trabalho às prioridades da organização, otimizar o fluxo de ponta a ponta, coordenar vários times e engajar partes interessadas. Os pesos são Mentalidade 28%, Liderança 25%, Produto 19% e Entrega 28%.

Como responder

✓ Conte com 120 itens em 3 horas (180 minutos), cerca de um minuto e meio por item; 20 são de pré-teste, não pontuados e não identificados, então trate todos com a mesma atenção.

✓ Há um intervalo de 10 minutos depois do item 60: revise as marcações da primeira metade antes de iniciá-lo, porque depois do intervalo não dá para voltar à seção anterior.

✓ Leia primeiro a última frase do enunciado (melhor ação, próxima ação, o que indica, qual técnica) e só depois o cenário, procurando o sinal que decide a resposta.

✓ Elimine primeiro as alternativas que contrariam o ágil: decidir pelo time, acrescentar aprovação ou documento, esconder informação, escalar sem antes tentar no time, punir ou procurar culpado.

✓ Entre duas boas, escolha a que ataca a causa no sistema e usa dados, em vez da que acelera ou contorna o sintoma.

✓ Em contas (Little, eficiência, CPI, SPI, EMV, ROI, payback, WSJF), escreva a fórmula antes de olhar as alternativas; os distratores são a razão invertida, o percentual isolado ou a conta sem a probabilidade.

✓ Nos itens de múltipla resposta e arrastar e soltar, confira quantas escolhas o enunciado pede e avalie cada opção isoladamente contra o cenário.

✓ Quando o enunciado cita um modelo pelo nome (Cynefin, Speed Leas, Kano, Tuckman, ADKAR, ROAM), responda com o termo exato do modelo e desconfie de distratores vindos de modelos vizinhos.

Erros que derrubam candidatos

✗ Escolher a alternativa mais controladora ou burocrática por parecer mais segura.

✗ Confundir lead time com tempo de ciclo e média com percentil ao responder sobre prazos.

✗ Tratar o ágil como sempre certo e ignorar quando o contexto pede abordagem preditiva ou híbrida.

✗ Considerar pronto ou demonstrável um item que não atende à Definição de Pronto.

✗ Usar velocidade ou pontos como meta, comparação entre times ou prova de valor.

✗ Errar o nível de conflito de Speed Leas ou o estilo situacional por não separar competência de motivação.

✗ Fazer a conta certa e escolher a interpretação errada (CPI abaixo de 1 é custo acima do previsto; eficiência baixa pede menos fila).

Na semana da prova

- ☐ Refaça os cálculos-chave até sair sem consultar: Lei de Little, eficiência de fluxo, CPI, SPI, EMV, ROI, payback, WSJF e canais de comunicação.
- ☐ Revise as listas fechadas que caem como nome: sequências do Cynefin, zonas de Stacey, valores do Scrum e do XP, Tuckman, Speed Leas, Thomas-Kilmann, Lencioni, Kano, ROAM e DORA.
- ☐ Faça ao menos uma prova completa cronometrada em 3 horas, com a pausa depois do item 60, para treinar ritmo e revisão da primeira metade.
- ☐ Revise os erros por domínio, dando atenção a Mentalidade e Entrega, que somam 56% dos itens.
- ☐ Na véspera, revise só os resumos de 'lembrar' de cada domínio e descanse; no dia, chegue cedo ou faça antes as verificações de sistema do exame on-line.
- ☐ Lembre que há até três tentativas dentro do período de elegibilidade de um ano: vá com calma, sem pressão de tudo ou nada.

REVISÃO FINAL

Folha de cola

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

1. Domínio I. Mentalidade

‣ Causa e efeito só ficam claros em retrospecto → **Cynefin complexo: sondar, perceber, responder**

‣ Sistema causando dano agora e ninguém sabe a causa → **Caótico: agir para estabilizar, depois investigar**

‣ Requisitos acordados, tecnologia nunca usada → **Stacey zona complicada: spikes e protótipos**

‣ Requisitos e tecnologia incertos → **Stacey zona complexa: abordagem ágil**

‣ Parte estável de prazo longo e parte incerta no mesmo programa → **Híbrido sincronizado por marcos comuns**

⚠ Analisar antes de agir é do domínio complicado; no caos o tempo de análise prolonga a exposição. Primeiro estabilize, depois investigue.

⚠ Sondar é para o complexo; numa rotina documentada com causa e efeito evidentes, variar o procedimento só introduz risco. Domínio claro: boa prática.

⚠ Na Stacey, isso é zona complicada; o problema está na tecnologia, então reduza a incerteza técnica com spikes e protótipos.

2. Domínio II. Liderança

‣ Influência por admiração, sem cargo nem expertise → **Poder de referência**

‣ Atender plenamente às preocupações das duas partes → **Colaborar (Thomas-Kilmann)**

‣ Prazo curto e metas de importância moderada → **Conciliar (meio-termo)**

‣ Ganhar virou mais importante que resolver; ataques pessoais → **Speed Leas nível 3: facilitação estruturada**

‣ Intenção de destruir o outro lado → **Nível 5 (guerra): separar as partes e intervenção externa**

⚠ Perguntas não ajudam quem não sabe: ensine o essencial primeiro, faça junto e retire o apoio aos poucos, depois volte ao coaching.

⚠ Delegar cedo expõe ao fracasso; quem ainda não tem competência pede direcionar, com orientação clara e acompanhamento.

⚠ A lacuna é de motivação, não de conhecimento: estilo apoiar. Direcionar soa como desconfiança e delegar reduz o contato que ela precisa.

3. Domínio III. Produto

‣ Dado / Quando / Então → **BDD**

‣ Cartão, conversa e confirmação → **3 Cs de Ron Jeffries**

‣ Valor perdido a cada semana de espera → **Custo do atraso (base do WSJF)**

‣ Dinheiro fictício para comprar funcionalidades → **Buy a feature**

‣ Cada stakeholder distribui 100 pontos → **Método dos 100 pontos**

⚠ Item fora da Definição de Pronto não é apresentado: dá falsa ideia de progresso. Mostre só o pronto e devolva o resto ao backlog.

⚠ A duração é fixa; estender quebra a cadência e adia a inspeção e a adaptação.

⚠ Fatias horizontais não entregam valor sozinhas: fatie na vertical, de ponta a ponta.

4. Domínio IV. Entrega

‣ Faixa do CFD que se alarga → **Gargalo / acúmulo nessa etapa**

‣ Só a faixa de entrada do CFD crescendo → **Demanda acima da capacidade: atuar na entrada**

‣ Burnup com escopo subindo quase tão rápido quanto o concluído → **Data em risco: negociar escopo**

‣ Velocidade subindo sem mais entregas → **Estimativas infladas, Lei de Goodhart**

‣ Rodadas anônimas de estimativa com discussão até convergir → **Wideband Delphi**

⚠ CPI = EV / AC, com EV = percentual de pontos concluídos x BAC; 375/800 é só o orçamento consumido e 1,25 é AC/EV.

⚠ SPI = EV / PV = percentual de pontos concluídos / percentual de sprints decorridas (30% / 40% = 0,75).

⚠ O denominador é sempre o lead time total (trabalho + espera): 9 / 32, cerca de 28%.

GABARITO

Respostas do diagnóstico

D1. A) Inspecionar o progresso rumo à Meta da Sprint em até 15 minutos → revise o capítulo 1

Por quê: A Daily Scrum é um evento de 15 minutos para os Developers inspecionarem o progresso em direção à Meta da Sprint e adaptarem o Sprint Backlog. Não é uma reunião de status para o Scrum Master, e discussões técnicas detalhadas acontecem depois, com quem for necessário.

✗ B) Incorreta. Não é reunião de status nem dura 30 minutos.

✗ C) Incorreta. O Product Backlog é tratado no refinamento, não na Daily.

✗ D) Incorreta. Detalhes técnicos ficam para depois da Daily.

> **Dica de prova:** Daily: 15 minutos, Developers, progresso rumo à Meta da Sprint.

D2. A) Criar condições para que más notícias sejam trazidas cedo, sem culpa, e refletidas no painel → revise o capítulo 1

Por quê: Na tipologia de Westrum, usada nas pesquisas do DORA, culturas generativas treinam os mensageiros, enquanto culturas patológicas os punem. Quando trazer más notícias é arriscado, os relatórios ficam verdes até o problema explodir. O praticante atua na causa, criando um ambiente em que riscos e atrasos são visíveis cedo e tratados como problemas do sistema.

✗ B) Incorreta. Mais indicadores do mesmo tipo podem ser maquiados da mesma forma; a causa é cultural.

✗ C) Incorreta. Mudar a cor padrão não muda o comportamento; o status continua refletindo o que as pessoas se sentem seguras para informar.

✗ D) Incorreta. Procurar culpados confirma o medo que causou o problema e incentiva mais ocultação.

> **Dica de prova:** Status sempre verde com problemas escondidos é sintoma cultural: torne seguro trazer más notícias cedo.

D3. B) Como atividade contínua, sem ser um evento → revise o capítulo 1

Por quê: O refinamento é uma atividade contínua de dividir, detalhar e estimar itens do Product Backlog, feita pelo Product Owner e pelos Developers conforme necessário. Não é um dos eventos do Scrum e não tem timebox definido pelo Guia. Ele mantém os itens do topo prontos para a Sprint Planning.

✗ A) Incorreta. É feito pelo PO e pelos Developers.

✗ C) Incorreta. O Guia não define refinamento como evento nem timebox.

✗ D) Incorreta. Acontece continuamente, não uma vez só.

> **Dica de prova:** Refinamento: atividade contínua, não evento do Scrum.

D4. B) Colaborar → revise o capítulo 2

Por quê: O modelo Thomas-Kilmann descreve cinco modos: competir, colaborar, comprometer, evitar e acomodar. Colaborar combina alta assertividade e alta cooperação e busca uma solução que atenda plenamente aos dois lados, geralmente o modo preferido em times ágeis quando há tempo e confiança.

✗ A) Incorreta. Evitar adia o conflito sem resolvê-lo.

✗ C) Incorreta. Competir privilegia um lado à custa do outro.

✗ D) Incorreta. Acomodar abre mão das próprias preocupações.

> **Dica de prova:** Resolver atendendo aos dois lados: colaborar.

D5. A) Pelo problema dos comerciantes que a iniciativa resolve e pelo resultado esperado → revise o capítulo 2

Por quê: Promover visão e propósito compartilhados começa por explicar por que a iniciativa existe: o problema de quem será atendido e o resultado esperado. A DORA destaca a visão clara como uma das dimensões da liderança transformacional. Com o porquê entendido, o time decide melhor o como e o quê.

✗ B) Incorreta. Papéis são necessários, mas sem um propósito compartilhado a divisão de responsabilidades reforça os silos entre áreas.

✗ C) Incorreta. Começar pela solução técnica pula o entendimento do problema e pode fechar cedo demais as opções.

✗ D) Incorreta. O cronograma responde quando, mas não explica por que o trabalho importa e tende a virar o foco principal da iniciativa.

> **Dica de prova:** Abertura de iniciativa: comece pelo porquê, o problema e o resultado, antes do como e do quando.

D6. B) Combinar no acordo de trabalho quem acionar, em que ordem e em quanto tempo para cada tipo de bloqueio → revise o capítulo 2

Por quê: Acordos de trabalho são normas compartilhadas sobre como o time trabalha, se comunica e colabora. Incluir neles um caminho de escalonamento, com quem acionar, em que ordem e com que prazo, evita respostas improvisadas e diferentes a cada bloqueio e acelera a resolução.

✗ A) Incorreta. Dá visibilidade, mas não define quem age nem em quanto tempo o bloqueio deve ser escalado.

✗ C) Incorreta. Concentrar tudo no gerente cria um gargalo e reduz a autonomia do time para resolver o que está ao seu alcance.

✗ D) Incorreta. Um lembrete genérico não define quem acionar nem em que prazo, e o comportamento tende a variar de novo.

> **Dica de prova:** Cada um trata o bloqueio de um jeito: defina o caminho de escalonamento no acordo de trabalho.

D7. B) Priorizar os habilitadores com as funcionalidades, por critério econômico e capacidade reservada → revise o capítulo 3

Por quê: No SAFe, habilitadores estendem a pista arquitetural ou melhoram o fluxo de desenvolvimento. Por produzirem resultados tangíveis, devem ser visíveis e tratados como os demais itens do backlog. Eles são priorizados junto com as funcionalidades, com critérios econômicos como o WSJF, e muitas vezes com uma alocação de capacidade combinada entre negócio e tecnologia.

✗ A) Incorreta. Trabalho escondido nas estimativas fica invisível e não pode ser priorizado nem acompanhado.

✗ C) Incorreta. Concentrar o trabalho técnico no fim do incremento mantém a lentidão por meses e acumula risco.

✗ D) Incorreta. Um backlog paralelo separa decisões que disputam a mesma capacidade e tira a visão do conjunto de quem prioriza.

> **Dica de prova:** Habilitadores ficam visíveis no mesmo backlog e disputam capacidade com as funcionalidades por critério econômico.

D8. C) Registrar cada pedido não planejado como cartão no quadro, com tipo próprio → revise o capítulo 3

Por quê: O guia da Kanban University observa que muito do trabalho nas organizações fica escondido e que visualizá-lo aumenta a transparência. Trabalho não planejado invisível distorce a percepção de produtividade e a previsibilidade. O primeiro passo é registrá-lo no quadro, com tipo próprio; depois, com dados, o time pode reservar capacidade ou criar uma classe de serviço.

✗ A) Incorreta. Exigir que tudo espere a próxima iteração pode atrasar pedidos realmente urgentes e é uma decisão tomada sem conhecer a demanda.

✗ B) Incorreta. Cortar o compromisso pela metade é um palpite; sem dados sobre a demanda escondida, o time não sabe quanto reservar.

✗ D) Incorreta. Um plantonista pode ser uma boa solução depois, mas sem visibilidade do volume não há como dimensioná-lo nem mostrar o problema à gerência.

> **Dica de prova:** Trabalho invisível não pode ser gerenciado: primeiro coloque as interrupções no quadro, depois decida como tratá-las.

D9. B) Registrar a versão mais simples como história e criar uma história para cada exceção → revise o capítulo 3

Por quê: O padrão simples/complexo orienta a perguntar qual é a versão mais simples da história quando ela cresce durante a discussão. Essa versão vira uma história com critérios de aceitação objetivos, e cada variação ou exceção passa a ser uma história separada. Assim o time entrega e aprende com o caso principal enquanto a PO decide quais exceções valem a pena e em que ordem.

✗ A) Incorreta. Agrupar os erros em um item técnico tardio cria uma fatia sem valor próprio e acumula risco no fim do release.

✗ C) Incorreta. Esperar o mapeamento completo mantém a história grande e difícil de estimar e atrasa o feedback do caso principal.

✗ D) Incorreta. As dúvidas são de regra de negócio, não de viabilidade técnica; um spike é o último recurso e não substitui a conversa com a PO.

> **Dica de prova:** História que cresce a cada 'e se': separe o caminho feliz e transforme cada exceção em uma história própria.

D10. A) Gráfico de burndown de riscos → revise o capítulo 4

Por quê: O risk burndown soma a exposição (probabilidade vezes impacto) dos riscos abertos em cada iteração e plota o total ao longo do tempo. Uma curva descendente mostra que as respostas aos riscos estão funcionando; uma curva estável ou crescente é um alerta visível para os patrocinadores.

✗ B) Incorreta. A matriz de probabilidade e impacto mostra a situação dos riscos em um momento, mas não a tendência ao longo das iterações.

✗ C) Incorreta. O burnup mostra trabalho concluído e escopo, não a exposição a riscos.

✗ D) Incorreta. O CFD mostra a quantidade de itens em cada etapa do fluxo, não os riscos.

> **Dica de prova:** Risk burndown: exposição total a riscos plotada a cada iteração.

D11. C) Analisar cada segmento separadamente, levantando as necessidades e os resultados esperados por ele → revise o capítulo 4

Por quê: O ECO pede que o praticante identifique e analise o cliente e suas necessidades. Quando um produto atende públicos diferentes, segmentar os clientes e entender as necessidades e os resultados esperados de cada segmento é o ponto de partida para priorizar com critério e para interpretar o feedback, que muitas vezes parece contraditório porque mistura grupos.

✗ A) Incorreta. Uma persona média não representa nenhum dos grupos reais e repete o erro de tratar públicos diferentes como um só.

✗ B) Incorreta. Somar pedidos de grupos distintos esconde as diferenças e favorece o segmento mais numeroso, que pode não ser o de maior valor.

✗ D) Incorreta. A área comercial tem uma visão de segunda mão; o time deve ouvir os segmentos diretamente.

> **Dica de prova:** Feedback contraditório costuma vir de segmentos diferentes misturados: segmente antes de priorizar.

D12. B) Análise pre-mortem → revise o capítulo 4

Por quê: No pre-mortem, o time assume que o projeto falhou e trabalha de trás para frente para descobrir por quê. A técnica reduz o otimismo excessivo, dá espaço para quem tem receios e gera riscos concretos para o backlog ajustado ao risco. Retrospectiva e revisão pós-implantação olham para o passado; o valor agregado mede desempenho de custo e prazo.

✗ A) Incorreta. Olha para uma release já concluída.

✗ C) Incorreta. Acontece depois da entrega, não antes.

✗ D) Incorreta. Mede desempenho de custo e prazo, não identifica riscos.

> **Dica de prova:** Imaginar o fracasso antes de começar: pre-mortem.

Agora, treine no formato real

Faça as 6 provas completas do curso PMI-ACP: 720 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 PMI (Project Management Institute). As questões são inéditas e não reproduzem a prova real.