Acesse a versão completa em: https://alessiojr.com/disciplines/gps/conteudo/00-livro/

De: O Mentor

*Scrum é uma abordagem útil, mas não é a única forma de organizar o trabalho. Neste capítulo, vamos conhecer o fluxo contínuo do **Kanban**, os princípios do **Lean** e práticas de engenharia associadas ao **Extreme Programming (XP)**. A escolha e a combinação dessas abordagens dependem do contexto, do tipo de demanda e das necessidades da equipe.*

Versão compilada em 20/9/2026 – Capítulo isolado do livro.

1 Kanban: Fluxo Contínuo

A Figure 1 introduz as alternativas de organização do trabalho. O **Kanban** nasceu na fábrica da Toyota e foi adaptado para a engenharia de software por David J. Anderson. Em contraste com a cadência de Sprints do Scrum, Kanban costuma organizar o trabalho em **fluxo contínuo**. No Scrum, o Sprint Backlog pode ser adaptado durante a Sprint.

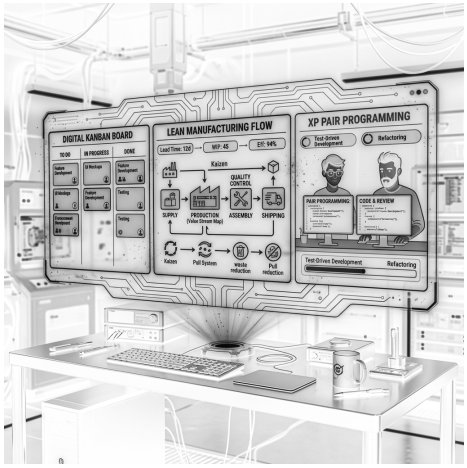


Figure 1: Ilustração de abertura: modelos ágeis e fluxo de trabalho. *Fonte: Imagem gerada por IA, com supervisão do autor.*

Essa abordagem pode ser útil para equipes de suporte, *DevOps* ou sustentação, nas quais as prioridades variam com frequência. Kanban não proíbe estimativas, planejamento, reuniões ou coordenação; a equipe define políticas explícitas adequadas ao seu fluxo. As demandas ficam visíveis em uma fila e podem ser puxadas quando há capacidade. Limites de **WIP (*Work In Progress*)** são uma prática frequente para tornar o trabalho em andamento visível e reduzir acúmulos.

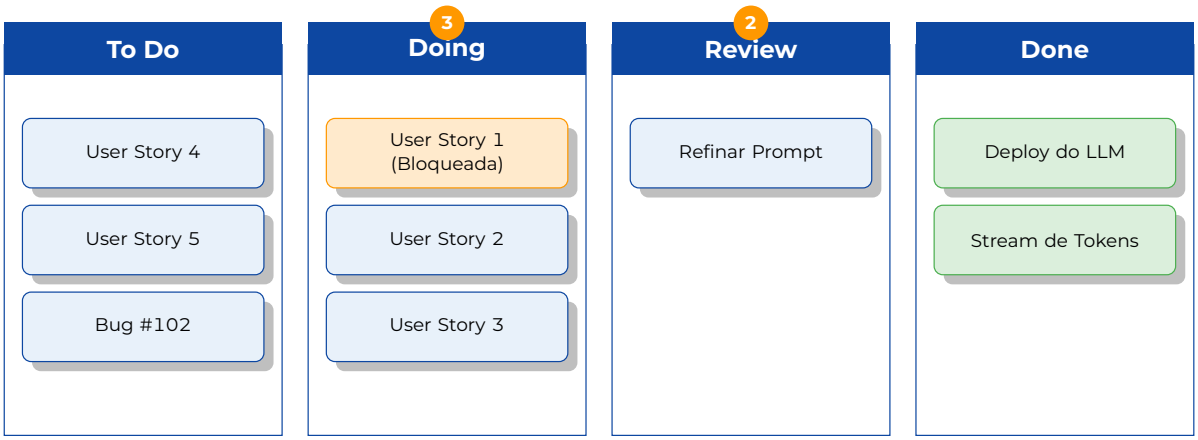


Figure 2: Quadro Kanban com políticas de limite de trabalho em progresso. *Fonte: Elaboração do autor.*

A Figura 2 sintetiza visualmente o conceito discutido nesta seção.

Se a política da coluna "Em Desenvolvimento" (*Doing*) definir limite 3 e já houver três cartões nela, a equipe normalmente evita puxar outro item até liberar capacidade. Ela pode, por exemplo, investigar bloqueios, revisar trabalho ou colaborar em um item em andamento (*swarming*). *Swarming* é uma estratégia possível, não uma obrigação do Kanban. O objetivo é favorecer a conclusão de itens sem ignorar políticas de serviço, urgências e o contexto do trabalho.

Versão compilada em 20/9/2026 – Capítulo isolado do livro.

Acesse a versão completa em: https://alessiojr.com/disciplines/gps/conteudo/00-livro/

Acesse a versão completa em: https://alessiojr.com/disciplines/gps/conteudo/00-livro/

• Teoria: No Mundo Real: Quando o Scrum Quebra

Imagine uma equipe responsável por uma API em produção. Um incidente crítico deixa o serviço indisponível. Em Scrum, os Developers podem adaptar o Sprint Backlog e agir diante da nova informação; não precisam aguardar a próxima Sprint Planning. A organização também pode definir políticas de resposta a incidentes e combinar Scrum com práticas de fluxo contínuo. Para equipes com alta urgência e variabilidade, Kanban pode tornar prioridades, capacidade e classes de serviço mais visíveis.

2 Lean Software Development: fluxo e desperdício

O **Lean** (Produção Enxuta) influenciou diversas abordagens de gestão e foi adaptado ao software, entre outros autores, por Mary e Tom Poppendieck. Ele oferece uma lente para observar fluxo, aprendizado e atividades que não contribuem para o resultado pretendido. A identificação e a redução de desperdícios (*Muda*) dependem do contexto e das necessidades do produto. Documentação, segurança, reuniões, arquitetura e conformidade não são desperdícios por definição: podem ser necessárias para reduzir riscos, coordenar decisões, atender regulações e manter o sistema. A equipe deve avaliar escopo, momento, custo e benefício dessas atividades. Entre os desperdícios discutidos no Lean de software, destacam-se:

- **Recursos Extras (*Gold Plating*):** Funcionalidades implementadas sem necessidade validada, que podem ampliar custo, complexidade e manutenção.
- **Trabalho Parcialmente Feito:** Trabalho ainda não integrado ou disponibilizado pode representar risco de perda, atraso de feedback e custo adicional. Isso não significa que seu valor seja necessariamente zero.
- **Espera:** O tempo em que o código fica aguardando revisão (*Code Review*) ou aprovação de um gerente.

O Lean nos ensina a adiar decisões irreversíveis até o último momento responsável (para termos dados empíricos) e a entregar valor o mais rápido possível ao cliente.

Versão compilada em 20/9/2026 – Capítulo isolado do livro.

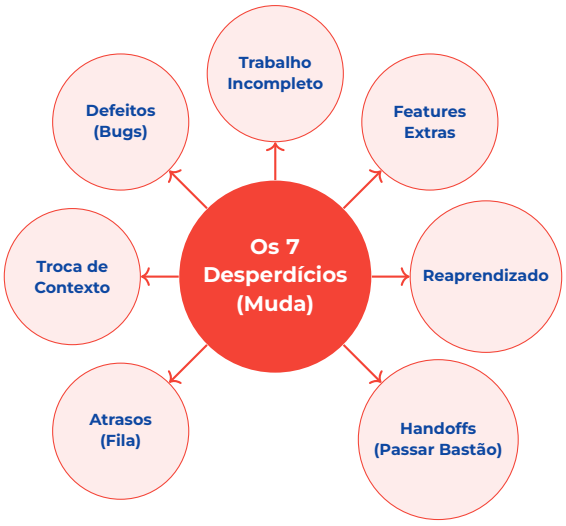


Figure 3: Sete desperdícios frequentemente discutidos no Lean de software. *Fonte: Elaboração do autor.*

A Figura 3 sintetiza visualmente o conceito discutido nesta seção.

▷ Exemplo: O Custo do Desperdício no Integrador

Imagine que a sua equipe demorou duas semanas para criar um sistema de "login social" (Google/Facebook), mas esqueceu de conectá-lo ao banco de dados, deixando o código parado em uma branch isolada e morta. No framework Lean, isso é um duplo desperdício: **Trabalho Incompleto** (o código não gerou valor e pode ficar obsoleto) e **Atrasos** (outras *features* vitais foram bloqueadas enquanto a equipe codificava o login inútil).

3 Extreme Programming (XP): práticas de engenharia

Enquanto Scrum define um framework para lidar com trabalho complexo, o **Extreme Programming (XP)**, associado a Kent Beck, reúne práticas de engenharia e colaboração. Equipes podem adotá-las e adaptá-las de acordo com seus riscos e restrições.

Entre as práticas frequentemente associadas ao XP estão:

- **Programação em Pares (*Pair Programming*):** Duas pessoas colaboram no mesmo problema, alternando papéis de *driver* e *navigator*. Pode ampliar revisão e compartilhamento de conhecimento, mas não elimina a necessidade de documentação adequada nem assegura ausência de defeitos.
- **Desenvolvimento Orientado a Testes (TDD):** Uma técnica na qual se escreve um teste antes do código de produção em ciclos curtos de falhar,

Versão compilada em 20/9/2026 – Capítulo isolado do livro.

Acesse a versão completa em: https://alessiojr.com/disciplines/gps/conteudo/00-livro/

passar e refatorar. TDD não é obrigatório para toda equipe e não garante cobertura total ou qualidade por si só.

- **Integração Contínua (CI):** Integrações frequentes acompanhadas de verificações automatizadas. A frequência e os controles devem ser definidos pela equipe e pela configuração do repositório.
- **Refatoração Contínua:** Melhoria incremental da estrutura do código, preservando seu comportamento observável e considerando testes, riscos e prioridades.

▶ Prática: No GitLab: ScrumBan no Projeto Integrador

Na vida real, não somos puristas de frameworks; somos pragmáticos. Para o Projeto Integrador (o nosso Assistente IA), adotaremos o **ScrumBan** (Scrum + Kanban) associado a práticas de XP: 1. **Regra da disciplina:** os *Milestones* quinzenais organizam as avaliações acadêmicas; não equivalem automaticamente a eventos Scrum. 2. **Prática da equipe:** o *Issue Board* do GitLab pode tornar o fluxo visível. A equipe pode experimentar limites de WIP e ajustar sua política conforme os dados observados. 3. **Controle técnico:** o pipeline pode executar verificações configuradas pelo projeto. Uma Issue deve ser fechada conforme critérios acordados, que podem incluir testes, revisão e documentação necessária; TDD é uma técnica possível, não uma exigência automática.

	Scrum	Kanban	XP	Lean
Ênfase	Framework de gestão	Fluxo e políticas explícitas	Práticas de engenharia	Fluxo de valor e aprendizado
Ritmo	Sprints de até um mês	Pode ser contínua ou definida pela equipe	Iterações e feedback frequentes	Depende do contexto
Equipe	Accountabilities Scrum	Papéis e reuniões definidos pela equipe	Colaboração técnica	Princípios aplicáveis a diferentes métodos
Práticas	Sprint Backlog e eventos	Quadro, políticas e possível WIP	TDD, pares, CI e refatoração	Mapeamento do fluxo de valor
Uso	Problemas complexos	Serviços e fluxos variáveis, entre outros	Contextos que se beneficiam de feedback técnico	Melhoria de fluxo e redução de desperdícios

4 Perguntas Reflexivas

Ajustando os parâmetros das metodologias alternativas, responda:

Acesse a versão completa em: https://alessiojr.com/disciplines/gps/conteudo/00-livro/

Versão compilada em 20/9/2026 – Capítulo isolado do livro.

1 Como um limite de WIP pode afetar filas, troca de contexto e tempo de fluxo? Indique as premissas da sua resposta.

2 Em que situações a programação em pares pode compensar seu custo? Quais evidências a equipe poderia observar?

3 Como testes automatizados, revisão e integração contínua podem reduzir riscos durante uma refatoração, sem eliminá-los completamente?

✓ Log: Assistente I.A.

Senhor, modelos operacionais alternativos engajados no banco de dados corporativo:

- **Kanban:** torna o fluxo e os gargalos visíveis. Limites de WIP e cadências são políticas que a equipe pode ajustar.
- **Lean:** ajuda a examinar desperdícios no fluxo. Atividades como documentação, segurança e conformidade devem ser avaliadas pelo valor e pelo risco que tratam.
- **XP:** oferece práticas como TDD, CI, programação em pares e refatoração. Elas reduzem riscos quando bem aplicadas, mas não substituem a avaliação de qualidade da equipe.
- **Diversidade Tática:** O engenheiro que só domina Scrum é como um cirurgião com um único bisturi. Kanban, Lean e XP são instrumentos complementares; a maestria está em saber qual usar em cada contexto.

★ Checkpoint do Projeto: Modelos Ágeis e Kanban

A equipe consegue explicar por que escolheu suas práticas e como avaliará seus efeitos?

- **Fluxo:** Se há muitos itens em andamento, registrem a situação e avaliem um limite de WIP ou outra política adequada.
- **Colaboração:** Programação em pares pode ser experimentada em tarefas complexas; registrem o resultado e decidam se ela é adequada ao caso.
- **Valor:** Reavaliem requisitos que perderam prioridade, preservando evidências e obrigações necessárias para o projeto.

Versão compilada em 20/9/2026 – Capítulo isolado do livro.

Acesse a versão completa em: https://alessiojr.com/disciplines/gps/conteudo/00-livro/

Agora que temos todo o conjunto de ferramentas conceitual das metodologias na mesa (Scrum, Kanban, Lean, XP e User Stories), estamos prontos para entrar no Capítulo 11 e dominar a complexa arte de Estimar o Inestimável e Priorizar o que entra no nosso funil.

Acesse a versão completa em: https://alessiojr.com/disciplines/gps/conteudo/00-livro/

Versão compilada em 20/9/2026 – Capítulo isolado do livro.