Metodologias Ágeis - Resumo

O que são Metodologias Ágeis?

Metodologias Ágeis são formas de organizar projetos, especialmente de software, com foco em entregas rápidas, colaboração constante, adaptação às mudanças e comunicação efetiva. Em vez de planejar tudo do início ao fim com detalhes imutáveis (modelo tradicional "cascata"), o ágil trabalha com ciclos curtos e feedbacks contínuos.

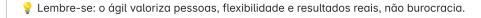
Imagine construir uma casa onde, a cada cômodo entregue, o cliente já pode visitar, testar e sugerir melhorias — ao invés de só ver o resultado final depois de meses. Isso é o espírito do ágil: **entregar valor continuamente**.

O Manifesto Ágil 🖉

Publicado em 2001 por 17 profissionais de software, o **Manifesto Ágil** define 4 valores e 12 princípios para o desenvolvimento ágil.

4 Valores:

- 1. Indivíduos e interações mais que processos e ferramentas.
- 2. Software funcionando mais que documentação abrangente.
- 3. Colaboração com o cliente mais que negociação de contratos.
- 4. Responder a mudanças mais que seguir um plano.



12 Princípios (resumidos):

- Satisfação do cliente com entregas contínuas.
- Mudanças são bem-vindas, mesmo tarde.
- · Entregas frequentes.
- Colaboração entre cliente e equipe.
- Projetos com indivíduos motivados.
- Comunicação cara a cara.
- Software funcionando é a principal medida.
- Ritmo constante.
- Excelência técnica e bom design.
- Simplicidade.
- Equipes auto-organizadas.
- Reflexão e ajustes frequentes.

Frameworks Ágeis: O que é o Scrum? 🖉

Scrum é um dos frameworks mais populares dentro das metodologias ágeis. Ele oferece um conjunto claro de papéis, eventos e artefatos que ajudam as equipes a serem mais produtivas, focadas e adaptáveis.

Scrum é como uma receita de bolo: fornece os ingredientes (papéis, eventos, artefatos) e o modo de preparo (como organizar o trabalho em sprints).

🧠 Dica para lembrar: "Scrum" vem do rugby, onde o time avança em conjunto. No ágil, todos avançam juntos também!

Papéis no Scrum 🖉

1. Product Owner (PO)

- Representa os interesses do cliente.
- Prioriza as tarefas no Product Backlog.
- o Define o que tem valor para o negócio.

2. Scrum Master

- Facilitador do processo Scrum.
- Remove impedimentos.
- Garante que a equipe siga o Scrum.
- o Treina e apoia o time.

3. Time de Desenvolvimento (Dev Team)

- Equipe multidisciplinar (devs, testers, designers, etc).
- Responsável por entregar incrementos de software funcionais.
- o Auto-organizado.



Artefatos no Scrum @

1. Product Backlog

- Lista priorizada de tudo que pode ser feito no produto.
- o Criado e mantido pelo PO.

2. Sprint Backlog

- Subconjunto do Product Backlog selecionado para uma Sprint.
- Inclui as tarefas necessárias para entregar esse incremento.

3. Incremento

• Resultado funcional e potencialmente entregável ao final de cada Sprint.

🍰 Analogia: o Product Backlog é o cardápio inteiro de um restaurante. O Sprint Backlog é o pedido da mesa. O Incremento é o prato pronto e servido.

Eventos no Scrum @

$1. \ \textbf{Sprint}$

- Período fixo (1 a 4 semanas) onde ocorre o trabalho.
- Deve entregar um incremento pronto ao final.

2. Sprint Planning (Planejamento da Sprint)

- o Define o que será feito na Sprint e como será feito.
- o Participam: PO, Dev Team e Scrum Master.

3. Daily Scrum (Reunião Diária)

o Duração: 15 minutos.

o Cada membro responde: o que fez ontem, o que fará hoje, se há impedimentos.

4. Sprint Review (Revisão da Sprint)

- Mostra o que foi entregue.
- Participam stakeholders e clientes.
- Serve para obter feedback.

5. Sprint Retrospective (Retrospectiva)

o Time reflete sobre o que funcionou bem, o que pode melhorar e define ações de melhoria.

Lembrete: A Sprint é o coração do Scrum. Todos os outros eventos giram em torno dela!

Por que usar Scrum?

- Entregas constantes com valor.
- Adaptação rápida às mudanças do cliente.
- Comunicação clara entre todos.
- Equipes mais engajadas e produtivas.

Dicas de Ouro para Memorizar 🔗

- PO é dono da visão.
- Scrum Master é o guardião do processo.
- Dev Team é quem constrói.
- Backlog é vivo, muda conforme aprendemos.
- Sprint é um mini-projeto completo.
- Daily é para alinhar.
- Retrospectiva é para crescer, não criticar.

Outros Frameworks Ágeis (menção honrosa) 🖉

- Kanban: baseado em fluxo contínuo. Usa cartões e colunas (To Do, Doing, Done).
- XP (Extreme Programming): foco técnico e qualidade de código.
- SAFe, LeSS, Nexus: para escalar o ágil em grandes empresas.