**Manifesto Ágil**

O **Manifesto Ágil** é uma declaração de princípios que fundamentam o [desenvolvimento ágil de software](https://pt.wikipedia.org/wiki/Desenvolvimento_%C3%A1gil_de_software" \o "Desenvolvimento ágil de software).

**Signatários Originais**

**Inicialmente, contou com dezessete signatários:** Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas.[1](https://pt.wikipedia.org/wiki/Manifesto_%C3%81gil#cite_note-agilemanifesto.org-1)

**O manifesto contém quatro valores fundamentais:**[**1**](https://pt.wikipedia.org/wiki/Manifesto_%C3%81gil#cite_note-agilemanifesto.org-1)

* Os indivíduos e suas interações acima de procedimentos e ferramentas;
* O funcionamento do software acima de documentação abrangente;
* A colaboração dos clientes acima da negociação de contratos;
* A capacidade de resposta à mudanças acima de um plano pré-estabelecido;

Não se trata, como poderia parecer à primeira vista, de um desprezo aos elementos e ferramentas tradicionais do desenvolvimento de software, mas sim do estabelecimento de uma escala de valores, na qual a flexibilidade e a colaboração são mais relevantes do que a rigidez de processos e planejamento clássicos.

**12 Princípios**

Os princípios do desenvolvimento ágil estão divididos em 12 princípios:[1](https://pt.wikipedia.org/wiki/Manifesto_%C3%81gil#cite_note-agilemanifesto.org-1)

* Garantir a satisfação do cliente entregando rapidamente e continuamente softwares funcionais;
* Softwares funcionais são entregues frequentemente (semanas, ao invés de meses);
* Softwares funcionais são a principal medida de progresso do projeto;
* Até mesmo mudanças tardias de escopo no projeto são bem-vindas.
* Cooperação constante entre pessoas que entendem do 'negócio' e desenvolvedores;
* Projetos surgem através de indivíduos motivados, e que deve existir uma relação de confiança.
* Design do software deve prezar pela excelência técnica;
* Simplicidade;
* Rápida adaptação às mudanças;
* Indivíduos e interações mais do que processos e ferramentas;
* *Software* funcional mais do que documentação extensa;
* Colaboração com clientes mais do que negociação de contratos;
* Responder a mudanças mais do que seguir um plano.

## O Scrum define uma abordagem para gerenciamento de projetos considerando um processo de desenvolvimento iterativo e incremental. Já o XP define uma metodologia ágil com foco nas atividades de desenvolvimento para pequenas e médias equipes.