

Manifesto for Agile Software Development¹

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

*That is, while **there is value in the items on the right**, we value the items on the left more.*

Comment: The Agile movement is not anti-methodology, in fact, many of us want to restore credibility to the word methodology. We want to restore a balance. We embrace modeling, but not in order to file some diagram in a dusty corporate repository. We embrace documentation, but not hundreds of pages of never-maintained and rarely-used tomes. We plan, but recognize the limits of planning in a turbulent environment.

February 11-13, 2001

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

see <http://agilemanifesto.org/>

Principles behind the Agile Manifesto

- Our highest priority is to **satisfy the customer** through early and continuous delivery of valuable software.
- **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.
- **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must **work together daily** throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and **trust them to get the job done**.
- The most efficient and effective method of conveying information to and within a development team is **face-to-face conversation**.
- **Working software** is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to **maintain a constant pace** indefinitely.
- Continuous attention to technical excellence and **good design** enhances agility.
- Simplicity--the art of **maximizing the amount of work not done**--is essential.
- The best architectures, requirements, and designs emerge from **self-organizing teams**.
- At regular intervals, **the team reflects** on how to become more effective, then **tunes and adjusts** its behavior accordingly.

see <http://agilemanifesto.org/>

From Chapter 6 of
Get Out of the Way!
How to Manage Development
of Timely, Innovative, and Relevant Products
by John V. Levy, Ph.D.