# Agile Manifesto - 4 Values

**Published on 30 January 2023 at 14:01**

**Author: Kimberly Wiethoff**

In the ever-evolving landscape of software development, agility has emerged as a guiding principle, revolutionizing the way teams collaborate, innovate, and deliver value to customers. At the heart of this transformation lies the Agile Manifesto, a seminal document that has shaped the ethos of modern software development since its inception in 2001. In this blog post, we embark on a journey into the Agile Manifesto, exploring its core principles and uncovering the profound impact it has had on the world of technology.

## The Birth of Agility:

The Agile Manifesto emerged from a gathering of seventeen software developers who convened in Snowbird, Utah, with the aim of finding common ground amidst diverse methodologies and philosophies. Frustrated by the rigid and bureaucratic nature of traditional software development processes, they sought a more flexible and responsive approach that prioritized individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan.



## The Four Core Values:

At the heart of the Agile Manifesto are four core values that encapsulate the principles of agility:

1. **Individuals and Interactions over Processes and Tools:** Agile emphasizes the importance of fostering a collaborative and empowered team environment, where individuals are valued for their unique contributions and interactions drive innovation and problem-solving.
2. **Working Software over Comprehensive Documentation:** Instead of getting bogged down in extensive documentation and bureaucracy, Agile advocates for a focus on delivering tangible results and functional software that meets the needs of customers.
3. **Customer Collaboration over Contract Negotiation:** Agile places a premium on engaging customers throughout the development process, seeking continuous feedback and iteration to ensure that the end product aligns with their evolving requirements and expectations.
4. **Responding to Change over Following a Plan:** In the face of uncertainty and complexity, Agile encourages adaptability and flexibility, embracing change as an opportunity for learning and improvement rather than viewing it as a setback.

## The Twelve Principles:

Complementing the four core values are twelve principles that provide practical guidance for implementing Agile methodologies effectively. These principles emphasize the importance of delivering value early and often, empowering self-organizing teams, fostering transparent communication, and embracing continuous improvement and reflection.

## The Impact of Agility:

Since its inception, the Agile Manifesto has catalyzed a paradigm shift in software development, empowering teams to embrace uncertainty, embrace change, and deliver value with unprecedented speed and efficiency. Agile methodologies such as Scrum, Kanban, and Extreme Programming have become ubiquitous in the tech industry, reshaping organizational structures, processes, and cultures.

## Individuals and Interactions over processes and tools

People are at the heart of software development. Improved communication and collaboration between software developers and software users is more important than the tools and processes used.  Software engineers create software for those who want it built and for those who want to use it to improve their lives and achieve their goals.  The processes and tools would ultimately be unable to deliver what is required if communication and collaboration were not present.

## Working Software over Comprehensive Documentation

While documentation is useful, it is more important to have working software. The agile methodology provides you with just enough documentation to begin developing the final product. As the software is developed, the delivered increment is evaluated, and additional documentation is created as needed.

The agile documentation is streamlined to provide just enough, but not too much, information to move forward. This avoids the lengthy delays caused by the documentation requirements of the Waterfall methodology. Technical specifications, technical requirements, design documents, test plans, and so on.

## Customer Collaboration over Contract Negotiation

Contracts were heavily relied on in the early stages of software development. A detailed product would be the subject of the contract. Most of the time, the end result would be a conflict between what the contract stated, what the product did, and what the customer required.

With Agile project management, the emphasis shifted to continuous development, allowing for a direct feedback loop with the customer. The product roadmap begins with a high-level concept and progresses through an iterative process to a detailed solution with multiple iterations that include a customer feedback loop along the way to ensure the product meets the customers' expectations.

## Responding to Change over Following a Plan

A static plan or roadmap was designed and documented in the early stages of software development, and development efforts had to adhere to a rigid plan. However, because software needs and requirements are constantly changing, as are priorities, this method of working was inefficient.

With a more flexible roadmap, the Agile Manifesto suggests that software development teams should be able to change direction and pivot when requirements change. A dynamic roadmap can be updated quarterly or monthly, allowing agile teams to adapt to the changing roadmap.

## Final Thoughts:

As we reflect on the Agile Manifesto and its enduring legacy, we are reminded of the transformative power of agility in driving innovation, collaboration, and customer satisfaction. By embracing the values and principles outlined in the Agile Manifesto, organizations can navigate the complexities of the digital age with confidence and resilience, unlocking new opportunities for growth and success. In the words of the Agile Manifesto, let us strive to "uncover better ways of developing software by doing it and helping others do it."

#AgileManifesto #AgileValues #IndividualsOverProcesses #WorkingSoftware #CustomerCollaboration #RespondToChange #AgilePrinciples #AgileDevelopment #AgileMindset #AgileTransformation