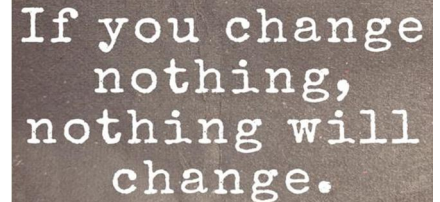# THE AGILE MINDSET

What Is An Agile Mindset?

Practitioners are thus said to have *an Agile mindset* when they are preoccupied—and sometimes obsessed—with innovating and delivering steadily more *customer* value, with getting work done in small self-organizing *teams*, and with collaborating together in an interactive *network*.

If you change nothing, nothing will change.

Such organizations have been shown to have the capacity to adapt rapidly to a quickly shifting marketplace.

By contrast, managers in traditionally run organizations are often said to have a **bureaucratic mindset** when they are primarily preoccupied with *making money for the company* and its shareholders, when they are organizing work according to *rules, roles and criteria* that they determine,

and when they are operating the organization as a *top-down hierarchy* with multiple layers and divisions.

What Is the Agile Manifesto?

The Agile Manifesto is a document that outlines the central values and principles of Agile software development. Officially referred to as **the Manifesto for Agile Software Development, the guide aims to provide an effective model for teams to successfully adopt the philosophy of Agile project management and use it to improve their work process**.

The lightweight framework of the Agile Manifesto was designed to improve upon existing software development processes, which were more complex and contained a lot of documentation. The founders wanted to speed up these processes and create a more efficient working model for teams.

Put simply, the Agile Manifesto is an alternative to traditional software development methodologies.

Although the Agile Manifesto was designed for the software development community, it has been used as a key resource for planning all types of projects. The State of Agile report noted numerous sectors outside software development that have adopted Agile practices, including IT, operations, marketing, HR, and sales.

But what exactly are the best practices outlined in the Agile Manifesto? Let's look at the values and principles in more detail.

The four Agile Manifesto values

The Agile Manifesto was founded on four key values, which are as follows:

. **Individuals and interactions over processes and tools**

. **Working software over comprehensive documentation**

. **Customer collaboration over contract negotiation**

. **Responding to change over following a plan**

1. Individuals and interactions over processes and tools

The first value gets straight to the core of Agile methodology: the focus is on the people. You can use the best processes and tools available to support your projects, but they will only work if your people are doing their best work. Your team is your most valuable resource. Communication plays a key role here — when people

interact with each other regularly and share their ideas, they build better products.

Jim Highsmith, one of the authors of the Agile Manifesto, wrote about the importance of the "mushy stuff," meaning prioritizing people over processes. Rather than simply proclaiming that people are the most important factor, he advises that software developers actually act as if people are the most important factor.

2. Working software over comprehensive documentation

Before Agile practices were fully implemented, teams would spend hours creating exhaustive documents with technical specifications, requirements, and more. These lists would be prepared before developers started to write the code, meaning the whole process was delayed as the documentation took so long to compile.

The Agile philosophy is to streamline these documents and condense the information into user stories. These stories equip the developer with all the details they need to start working on the software and get it ready for release.

The idea is to accelerate the launch process and make product tweaks in the early stages, improving the software in future iterations.

3. Customer collaboration over contract negotiation

The third value of the Agile Manifesto highlights the **importance of customer collaboration**. This is viewed as superior to contract negotiation, which involves **outlining product requirements with the customer before commencing the project and then renegotiating these deliverables at a later stage**. The issue here is that the customer is not engaged throughout the development cycle,

so teams lose out on user feedback that could enhance the product.

When you bring your customers into the development process, you can ask for their opinions regularly and take their suggestions on board. **By delving into their specific needs while the software is still being built, developers can gain valuable insights to create the ultimate user experience.**

4. Responding to change over following a plan

Traditional methodologies advocated for as little change as possible, recognizing that significant alterations could cost time and money. The aim was to create a comprehensive plan that followed a structured, linear path and avoided obstacles where possible.

The Agile mentality turns this rigid method on its head, arguing that change can benefit the software development

process. **When you embrace change, you open yourself up to new possibilities and ways to improve**. **Agile teams work in short, iterative cycles, meaning they can react quickly and implement changes on a continuous basis**. This ultimately leads to better products.

These four values form the basis of Agile software development, highlighting key priority areas for teams to

focus their energies on. The core values are supported by the

12 Agile Manifesto principles.

The 12 Agile Manifesto principles

12 Principles of Agile

- Satisfy the customer
- Measure of progress through working project
- Promote sustainable development
- Continuous attention to technical excellence
- Simplicity is essential
- Self-organizing teams
- Regularly reflect on continuous improvements
- Face-to-face conversation
- Motivated individuals
- Collaborate daily
- Deliver working software frequently
- Welcome changing requirements

The Agile Manifesto lists 12 principles to be followed by software developers:

These principles are as follows:

1. Place customer satisfaction above all priorities.

2. Self-organizing teams are the ideal team.

3. Working products are the best success.

4. Embrace logical change at every step of a project.

5. Face-to-face communication over other forms of communication.

6. Never underestimate the power of simplicity.

7. Work towards a sustainable pace of development.

8. It's essential for business and tech people to work together.

9. Put a sufficient amount of time into producing quality products.

10.   Aim to improve your working deliverables at a continuous rate.

11.   Learn valuable lessons from your failures and successes.

12.   Produce quality results by building projects around motivated teams.

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

What is the number one rule in software development? Keep your customer happy. Aim to deliver software to your users at regular intervals throughout the project life cycle, rather than keep them waiting for one final deliverable.

2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Software developers should be capable of handling last-minute changes to a project. They should be flexible enough to turn these changes into improvements quickly, minimizing delays and ensuring an uninterrupted flow of work.

3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Agile teams break large projects down into short timelines to guarantee regular delivery. In the Scrum methodology, these timelines are known as sprints, which often run between one and four weeks long.

4. Business people and developers must work together daily throughout the project.

As mentioned, collaboration is key to Agile project management. When the project stakeholders communicate on a daily basis, it minimizes the risk of confusion and ensures that everyone's goals remain aligned.

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

When you have the right people for the task, the project is more likely to be successful. Spend time choosing the perfect team, equip them with the resources they need, and trust them to deliver exceptional results.

6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Barriers are broken easily when teams can converse in person. Co-locate teams where possible to promote good communication and boost the flow of information. In a remote working situation, Zoom is a great alternative to

phone or email, enabling people to interact more effectively via camera.

7. Working software is the primary measure of progress.

To keep your customers satisfied, you must deliver high-quality software. That is your ultimate priority and your metric for success — everything else is a secondary consideration.

8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Agile teams must be consistent throughout their project life cycle, maintaining their speed throughout. This means they can sustain a constantly evolving environment without succumbing to delays.

9. Continuous attention to technical excellence and good design enhances agility.

In Agile, you don't just deliver one good product and call it a day — your focus should be on continuous improvement and innovation. Each iteration should bring with it a new update, feature, or advancement of some kind.

10. Simplicity — the art of maximizing the amount of work not done — is essential.

The Agile approach is to not overcomplicate things: meet your requirements and do just enough to complete the task. Don't waste time with additional steps that don't add any real value to your product.

11. The best architectures, requirements, and designs emerge from self-organizing teams.

Why micromanage teams when they are skilled enough to work on their own? By allowing them to work within their own structures, you create more room for ideas to flourish, ultimately delivering better results.

12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

When you review team performance regularly, you can spot issues before they escalate, as well as potential areas for improvement. A healthy Agile team is one that self-reflects in order to eliminate unhelpful practices and advance skills.

By studying the four values and 12 principles of the Agile Manifesto, software development teams will have a clear understanding of what it truly means to be Agile.

Why is the Agile Manifesto important?

The Agile Manifesto is a valuable resource for software development teams as it equips them with a flexible framework to guide their project management processes and uphold Agile best practices.

The foundational document also clarifies what is important to Agile project management, enabling teams to prioritize their activities and align their objectives. For example,

software developers will know that customer satisfaction is key, and so all project plans can be based around that guiding principle.

However, teams should be wary of what is known as the "Agile Industrial Complex." This refers to people and organizations that impose Agile best practices on teams rather than giving them the autonomy to decide what works best for them. There is no one-size-fits-all approach to

project management, and trying to force an unsuitable methodology will not deliver the desired results. Martin Fowler, one of the Agile Manifesto founders, adds that "even the Agile advocates wouldn't say that Agile is necessarily the best thing to use everywhere."

In short, the Agile Manifesto is a hugely important document for teams that wish to implement or sustain an Agile methodology in their organizations.

How to use the Agile Manifesto

How can you apply the values and principles of the Agile Manifesto to your work process? Rather than a detailed list of instructions or how-to guide, there is only one key nugget of advice to keep in mind: interpret it for your own team.

Discuss the Agile Manifesto with your colleagues and see what inspiration they draw from the values and principles. It is a flexible document designed to be adapted for your Agile

needs. Take the time to assess it, see what works, and figure out how you can adjust it for your team without losing sight of its core mission.

Just like your chosen project management platform, a guiding document such as the Agile Manifesto should be versatile enough to suit a variety of software development needs. **Tailor it to suit you and make it work for your Agile projects.**