

# Agile Software Development Demystified

Software development has been an established commercial field for several decades. For much of that time, software companies built software the same way other companies built physical products.

Then Agile came along. Agile development is an approach that has made waves in software project management, and has arguably improved the delivery of usable software across multiple industries.

## What is Agile Software Development?

Simply put, Agile software development is a “lightweight” approach to building software. It aims to make the development process efficient, dynamic and flexible – hence the term “Agile”!

Teams utilizing the Agile approach aim to operate in the following ways:

- They primarily value individuals and the interactions between them
- They aim to build working software as quickly as possible
- They collaborate with customers, making extensive use of feedback to improve the software
- They respond dynamically to changing project situations or specifications as they evolve

## Why Agile Software Development Was Born

The Agile approach was formed with the intention of ensuring customer satisfaction by delivering usable software as quickly as possible. The aim of Agile is to rapidly come up with a flexible software solution that can be changed quickly in response to customer feedback, and if bugs or glitches are found.

Instead of developing for long stretches of time in the hope of a perfect product launch, the Agile approach emphasizes the production of working software as quickly as possible, which is followed by multiple revisions and improvements. The Agile philosophy holds “working software” as its primary aim.

Agile software development also includes guidelines as to how developers should work together. It encourages face to face conversation and sustainable development that maintains a fairly even pace, without a stressful sprint-to-the-finish before a big deadline.

## How Does Agile Software Development Differ from Traditional Processes?

Traditionally, software development followed a process not far removed from the way in which a company might develop a physical product. There would be the classic sequential phases of Conception, Design, Implementation and Maintenance. This approach has proven itself in many traditional industries where the end product is unlikely to need much revision or modification.

However, as became apparent over the years, software is not such a product! It often needs multiple revisions, and customers are able to receive updates and improvements fairly painlessly. Software that is shipped with unforeseen problems or bugs and can't be fixed quickly is a major competitive disadvantage.

As such, the Agile process was created in order to allow developers to maintain continuous and dynamic improvement of their software, while it is in the hands of their customers.

## **Are there any Downsides to Agile Software Development?**

Agile has proven to be a very useful process for many software companies - particularly those that offer web, mobile or desktop applications.

However, it isn't always the perfect fit – it can be somewhat unwieldy in large organizations, especially where several different departments are involved in the creation of the software, such as in console game design.

Agile is also a poor fit for software where bugs have high downside, and risk of catastrophe - critical software systems in banking, aerospace and blockchain should almost certainly not be built with Agile.

In such systems, a traditional “waterfall” development approach may be a better fit.

That said, Agile has taken software development by storm. It is the preferred choice of many software companies, and is even having an impact on the field of project management outside of the software industry. It is truly a real leap forward, and will certainly play a central role in how software gets made in the years to come.