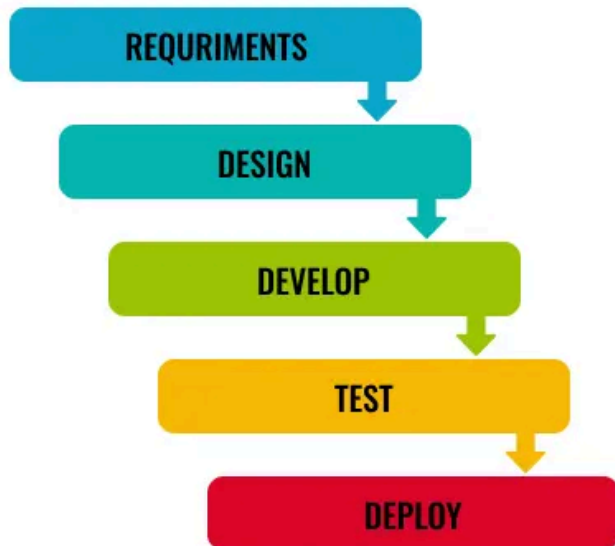




# Agile vs. Waterfall: Choosing the Best Project Management Strategy

## WATER FALL



VS



As a project manager for a digital product or a software developer for a SaaS project, you must have somehow known the ongoing debate about Agile vs. Waterfall. The theoretical explanations may bog you down. Let us not talk about that, however, it is crucial you know the two best project management aspects that create a phenomenal impact on any company's progress.

Knowing which is the best – Agile or Waterfall is actually a big deal. For any digital project management, you must first examine these core methodologies as they will help you finalize one. Also, a lot depends on the kind of project you have on hand. Let us find out some key differences between Agile and Waterfall. We will also understand when to use these technologies and when you can even combine them in a project.

## Key Difference Between Agile and Waterfall

Agile is forever adaptive to changes and delivers work in iterations and increments. Whereas, Waterfall focuses on planning all the requirements first, gathers resources, and begins with the project. This leaves very little room to change or become flexible thereafter.

Waterfall projects just begin from the start and end up at the finish point, adhering to the structured process and deliverable schedules. In contrast, Agile project management does not define any prescribed list of requirements at the initial stage. The team is not bound to practice documentation and follow a structured pattern. Rather, the team works on adding value to the customer's needs.

## Waterfall Methodology



It is one of the most traditional methodologies applied in software development. It is a linear methodology, consisting of well-defined phases that include planning, design, development, testing, and deployment. Just like a waterfall, the development process flows according to the phases defined. One cannot go back and make changes. This method is suitable for projects having predefined stages and does not need any iterations in the middle.

However, the real world is different. This method is often rigid and is considered an old-school approach. To take the benefit out of this technique, the only way is effective communication during each phase and Waterfall methodology can become a perfect hit.

## Pros

- Waterfall methodology prefers gathering requirements upfront. The team of developers and clients finalize the deliverable upfront and simply follow the structured plan and design. They measure progress against the key deliverables list in the ongoing project.

All this is comprehensively documented, making it easier for the new programmers to implement easily.

- The entire project is divided into different sequential phases that are executed one after the other. Through a structured workflow, it becomes easier to measure the progress and find out loopholes against the well-defined milestones structured at the very beginning of the project.
- A straightforward testing process is implemented, referring to the specific use cases in the past and as defined in the functional specifications at the end of the product development phase.
- Clients and stakeholders have minimum involvement except for product reviews, approvals, and status meetings.

## Cons

- All the stakeholders are involved meagerly to provide feedback in the earlier stage of the project life cycle. This hampers the possibilities to know whether the product would generate customer satisfaction or not. This limits the time as well as the budget to iterate as per the customer feedback.
- This methodology is not flexible enough to apply learnings to the ongoing project phase and make the necessary course of action. Hence, the reality differs from the planned action. Waterfall nature itself contradicts the adaptability principle. So it is difficult to make changes during the execution even if they are found to be necessary.
- The delayed testing phase might hamper identifying issues.
- The well-structured nature might prolong the project duration and delay the time-to-market gaps

## Use Cases of Waterfall

- Software development projects for which requirements are previously well-defined.
- Event planning especially when...

Read More of this blog [here...](#)