International Journal of Multidisciplinary Research and Growth Evaluation.

# Agile and waterfall: Two different ways of approaching software projects

**Syeda Hajira Kawsar**
Data Engineer, Independent Research, USA

* Corresponding Author: **Syeda Hajira Kawsar**

## Article Info

**ISSN (online):** 2582-7138
**Volume:** 04
**Issue:** 05
**September-October 2023**
**Received:** 13-08-2023
**Accepted:** 15-09-2023
**Page No:** 1121-1124

## Abstract

This paper focuses on the two major software project management processes, Agile and Waterfall, and explains the fundamental concepts, advantages, and disadvantages of each process. Agile is famous for its incremental and adaptive style that helps teams adapt to changing conditions that are existent in project contexts. On the other hand, there is a Waterfall approach, which is a more rigid framework applicable for projects that are expected to undergo little change in requirements. This paper compares the various methodologies in terms of systems used, functioning, uses and the aspects that determine their use. Drawings and illustrations, such as flowcharts and diagrams, are provided to support the concepts explained and presented. The authors also describe the conditions suitable for applying each of the methodologies under discussion.

**DOI:** https://doi.org/10.54660/.IJMRGE.2023.4.5-1121-1124

**Keywords:** Projects, software development methodologies, waterfall methodology, agile methodology, iterative model, sequential model, risk management

## Introduction

The process of developing software has become more challenging as the projects are complex, and people need effective tools to solve the problem of project management, including time, cost, and quality [1]. There is a wide variety, but the most significant difference can be divided into two categories called Agile and Waterfall. Agile processes are always encouraging and incorporate change and iterative improvement while Waterfall requires both fixation on and adherence to phases. Such methodologies are aimed at different kinds of projects, teams, and client expectations, so it is critical for them to be familiar with such approaches. Agile or, more accurately, agile software development started in the early 2000s in response to the inadequacies of a traditional sequential approach known as the Waterfall model [2]. It lays stress on cooperation, flexibility, and small incremental steps, which is more suitable for the change-oriented project. The latter, known as Waterfall, originates from engineering and manufacturing and was introduced to SW development in the 1970s. It imposes an order on the process such that the process of requirement gathering is done before the design phase and so on; there is no ambiguity about what must be accomplished or when.

The unfolding of this paper seeks to explore the methods of Agile and Waterfall, the strengths of the two approaches and the weaknesses that comes with each. By the end of this comparative analysis, readers will develop a better understanding of how each of these methodologies can be best deployed when managing specific kinds of projects.

## Basic Concepts of Agile / Waterfall

### Agile Principles

- Agile methodologies are guided by the Agile Manifesto, which outlines four core values [3]. These includes:
- Individuals and interactions over processes and tools: It stressed the cooperation and interaction of its members.
- Working software over comprehensive documentation: Working on sensitive and urgent deliverables that might be important to the customer [4].
- Customer collaboration over contract negotiation: Stakeholders are frequently contacted to ensure goal congruence of all the projects being carried out.

- Responding to change over following a plan: The ability to modify requirements if need arises.

Scrum is a format of Agile that works in cycles called sprints that range between 2 weeks and 4 weeks [5]. Instead of long elaborate development cycles, which can cause programs to become skewed from their intended use, these cycles allow for constant enhancement, steady, gradual release of working software, and customer satisfaction. Scrum, Kanban or Extreme Programming (XP) are methodologies that teams choose for putting Agile into practice.

### Waterfall Principles

**Waterfall follows a linear, sequential process with clearly defined phases:**

- Requirements gathering: Photocopying all aspects of a project before commencement to ensure that should a problem arise, all the specifications have been put in place.
- Design: Development of explicit system designs and specifications [5].
- Implementation: Implementation of the design by coding and development.
- Verification: Checking a product so as to confirm that it can meet specific requirements.
- Maintenance: Interactive inputs in terms of maintenance and upgrade services after the project has been implemented [6].

The rigid structure provides the much-needed planning and documentation to leave little room for interpretation once the project is initiated.

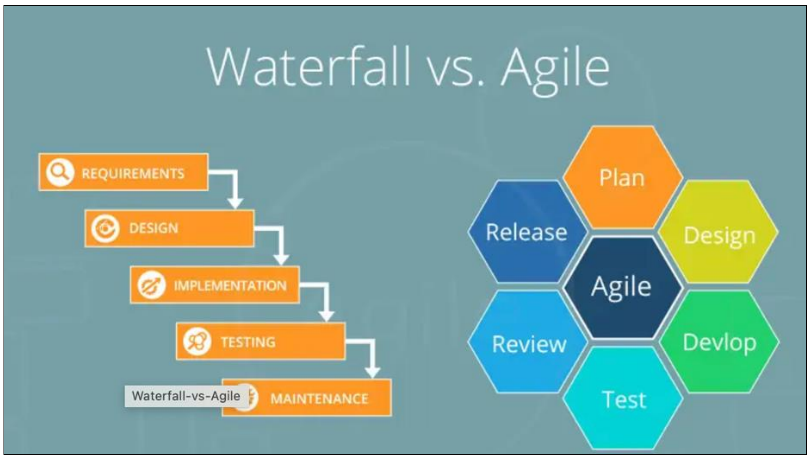


**Fig 1:** Waterfall vs Agile
(Source: www.nimblework.com) [14]

### Work flow and Appearance

### Agile Workflow

Agile is a cyclic or spiral approach to operation. Sprints involve planning and estimation, development, testing and integration and finally review by the stakeholders [7]. A simplified flowchart of the Agile workflow is as follows:

- Backlog prioritization
- Sprint planning
- Iterative development
- Sprint review and retrospective
- Usage of feedback in the next sprint

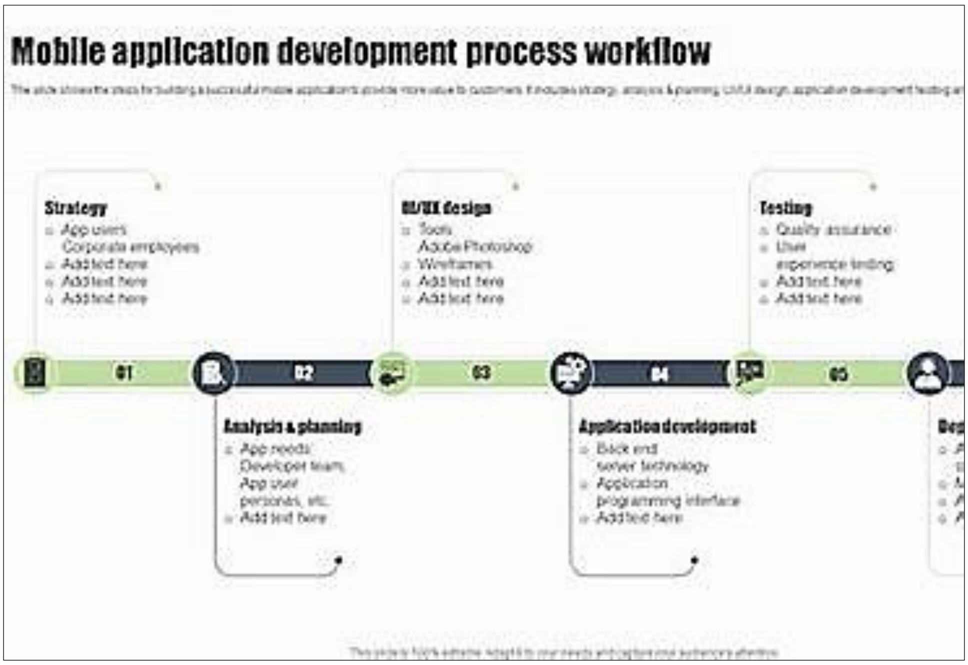
One important piece of a team's communication process is progress tracking using visual aids such as the Kanban board or burndown chart. These tools increase transparency and make everyone on the team on the same page.

### Waterfall Workflow

As it is clear, Waterfall is sequential in its execution process; the phases of the project are strictly integrated and successive [8]. A typical flowchart of the Waterfall process includes:

### Requirement analysis

- System design
- Implementation (coding)
- Testing and validation
- Deployment



Mobile application development process workflow

The slide shows the steps for building a successful mobile application to provide more value to customers. It includes strategy, analysis & planning, UI/UX design, application development, testing an

01 Strategy
- App users Corporate employees
- Add text here
- Add text here
- Add text here

02 Analysis & planning
- App needs: Developer team, App user personas, etc.
- Add text here

03 UI/UX design
- Tools Adobe Photoshop
- Wireframes
- Add text here
- Add text here

04 Application development
- Back end server technology
- Application programming interface
- Add text here

05 Testing
- Quality assurance
- User experience testing
- Add text here
- Add text here

Dep

This slide is 100% editable. Adapt it to your needs and capture your audience's attention.

**Fig 2:** Mobile Application Development Process Workflow
(Source: www.slideteam.net) [15]

### Maintenance

The initial aspect of project planning is the creation of Gantt charts in order to monitor the activities and their relations in the process of the Waterfall project. These charts can help give an accurate view of the project and its due dates and accomplishments so planning is more accurate [9].

## Comparative Analysis

### Flexibility and Adaptability

Agile is all about handling changes as it permits this by late in the developmental cycle. For example, shifts in the choice of measurable attributes of the product can be caused by customer reviews while not compromising the project. This versatility means that Agile is especially well suited for high turnover industries such as technological and media fields. Still, its approach is rather fixed, as the changes in the process require a transition through all the phases of the Waterfall [10]. This limitation is normally an issue of worry in fluctuating or unpredictable project forums.

### Risk Management

Since it is an iterative approach, for instance, it also entails that risks are detected and controlled at an early stage. The testing during sprints helps the teams sort out problems one at a time. For example, a defect that is found in one sprint can be fixed before the start of another sprint, thereby minimizing instances of error propagation. Problems in Waterfall, however, are rooted in the belief that risks can be avoided when elaborate planning is done. The problem with this approach is that if risks are identified at a later stage in project development, they could be expensive to manage. That is why Waterfall lacks intermediate deliverables that could help manage these problems more effectively.

### Cost and Time Efficiency

Agile reduces wastage for the simple reason that only the most important aspects of a project are worked on, and even then, only a little at a time. However, due to its recursive approach, the application of the framework might elongate schedules in some instances, where massive changes are needed. Waterfall, with its sharp and predictable division of time and cost, is most advantageous for projects with nondynamic needs, but can be immensely unprofitable in the case of changes [11]. For example, a waterfall project may require an overhaul when the client's requirements change in the middle of project implementation.

### Stakeholder Involvement

In this situation, agile focuses on the involvement of stakeholders during the execution of the project and the expectations that are to be followed. Sprint review is a cycle that involves a number of stakeholders who contribute to the subsequent stages of the project, as per their feedback. The waterfall approach engages stakeholders mostly during the requirements collection phase and at the project's end. This limited interaction can make some goals unsuitable If the requirements are misunderstood in the early model or change in the developing phase.

## Real-World Applications

### Agile Use Cases

### Agile is ideal for

- **Startups:** Ideation and customer feedback cycles are valuable activities for startups due to the ability to create original prototypes and improve or make changes quickly.
- **Dynamic projects:** Projects with changing parameters, such as the development of a mobile application or cloud solutions.
- **Collaborative environments:** Teams who invest major efforts in searching for the ways to communicate and achieve flexibility.

### Waterfall Use Cases

### Waterfall excels in

- **Regulated industries:** Such sectors as healthcare and finance that require code compliance and strict documentation measures.

- **Stable projects:** Projects which can have easily and clearly described requirements and demands that remain constant, for example, the construction of power plants [12].
- **Predictable environments:** Situations in which it is essential not only what is being discussed but also how it is structured.

### AS Agile and Waterfall Software Development Approaches: Technological Tools

In Agile, there are useful things as Jira, Trello, and Slack that serve as collaboration and progress trackers. In Waterfall, tools such as Microsoft Project and Primavera also guarantee proper planning and implementation. These technologies increase flexibility and productivity within their approaches to the methodologies.

### Case Study Examples

An Agile project may be a project aimed at creating a mobile application and its features are improved incrementally according to the users' feedback. However, a waterfall project might involve erecting a bridge where planning guarantees that the structure will stand and meet standards [13].

### Conclusion

Agile and Waterfall are basically working models that need to be selected based on the particularities of a project itself – its scope and requirements. This is because Agile has frequent updates as it is used in application development projects that require the incorporation of changes that meet customers' needs, while Waterfall is consistent in its process because it's used for projects whose activities are predetermined. This paper aims to establish how project managers can meet organizational goals and objectives through appreciating the strengths and weaknesses of each methodological approach in the organization.

Based on considerations of projects' characteristics and team and stakeholders' expectations, it is possible to use the strengths of both Agile and Waterfall approaches. The various strategic alignments are also operational since they improve efficiency and minimize risks while providing a guarantee that the end products will satisfy those expectations. In conclusion, 'the acquisition and utilisation of these methodologies' are tools that are most beneficial when applied as decision–making strategies in efficient software development and project management.

### References

1. Andrei B-A. A study on using waterfall and agile methods in software project management; c2019. Available from: https://web.rau.ro/websites/jisom/Vol.13%20No.1%20-%202019/JISOM-SU19-A12.pdf
2. Senarath U. Waterfall methodology, prototyping and agile development. 2021. DOI: https://doi.org/10.13140/RG.2.2.17918.72001
3. Dima AM, MAASSEN MA. From waterfall to agile software: Development models in the IT sector, 2006 to 2018. Impacts on company management. J Int Stud. 2018;11(2):315-325. Available from: https://www.ceeol.com/search/article-detail?id=718102
4. Mokhtar R, Khayyat M. A comparative case study of waterfall and agile management. SAR J Sci Res. 2022;5(1):52-62. Available from: https://www.ceeol.com/search/article-detail?id=1028934
5. Säisä M, Tiura K, Janne R. Waterfall vs. agile project management methods in university-industry collaboration projects. 2018. Available from: https://www.theseus.fi/handle/10024/153443
6. Kodmelwar DMK, Futane DPR, Pawar PSD, Lokhande PSA, Dhanure PSP. A comparative study of software development waterfall, spiral and agile methodology. J Posit School Psychol. 2022;6(3):7013-7017. Available from: https://www.journalppw.com/index.php/jpsp/article/view/4267
7. Thummadi BV, Lyytinen K. How much method-in-use matters? A case study of agile and waterfall software projects. J Assoc Inf Syst. 2020;21(4):864-900. DOI: https://doi.org/10.17705/1jais.00623
8. Zahidul A, Ferworn A, Islam Z. A comparison between agile and traditional software development methodologies. 2020. Available from: https://core.ac.uk/download/pdf/539593566.pdf
9. Alzeyani EM, Szabó C. A study on the methodology of software project management used by students whether they are using an agile or waterfall methodology. IEEE Xplore. 2022 Oct 1. Available from: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9974749&casa_token=35GCTMg0NbMAAAAA:oPoKhfVJhmSHPKqQ6-xM06qkRxMoO_HCQ_WWwId6SssLK0RYXJYot87Io1XOR7A1VOjBslBaqwM
10. Demirağ A, Nur E, ÜNAL C. Analysis and comparison of waterfall model and agile approach in software projects. AJIT-e: Online Acad J Inf Technol. 2023;14(54):183-203. DOI: https://doi.org/10.5824/ajite.2023.03.002.x
11. Fagarasan C, Popa O, Pisla A, Cristea C. Agile, waterfall and iterative approach in information technology projects. IOP Conf Ser: Mater Sci Eng. 2021;1169(1):012025. DOI: https://doi.org/10.1088/1757-899x/1169/1/012025
12. Sinha A, Das P. Agile methodology vs. traditional waterfall SDLC: A case study on quality assurance process in software industry. IEEE Xplore. 2021 Sep 1. Available from: https://ieeexplore.ieee.org/document/9614779
13. Thesing T, Feldmann C, Burchardt M. Agile versus waterfall project management: Decision model for selecting the appropriate approach to a project. Procedia Comput Sci. 2021;181(1):746-756.
14. Singh M. Moving from waterfall to agile with Kanban. NimbleWork. Dec 2022. Available from: https://www.nimblework.com/blog/waterfall-to-agile-with-kanban/
15. Mobile application development process workflow. 2023 May. Available from: https://www.slideteam.net/mobile-application-development-process-workflow.html