AGILE VS. WATERFALL: A COMPREHENSIVE ANALYSIS OF METHODOLOGIES FOR EFFECTIVE PROJECT MANAGEMENT

Amejuma Ebule

MSC, Business Analyst

Business intelligence and Analytics

Business Analysis and Artificial Intelligence, Droyslden, England

Abstract: The following paper discusses Agile and Waterfall approaches in project management, the purpose, benefits and drawbacks of their implementation in software development and in other fields. The Agile process, which is flexible and repetitive in nature is helpful when project environment is convoluted and changing all the time. In contrast, the Waterfall methodology is a straightforward, sequential approach that offers structured planning and predictability—ideal for projects with well-defined requirements. Understanding those methodologies' origins, approaches, and examples of usage through cases, this work reveals the differences: flexibility, documentation, and management of stakeholders' interactions. A combination of both methodologies comes out as a viable solution to meet various project requirements. The results should help the project managers to choose correct methodologies depending on the project characteristics and to improve the processes, resources, and results.

Keywords: Project Management Methodologies, Agile, Waterfall, Software Development, Iterative Planning.

I. INTRODUCTION

This study aims to provide a comprehensive comparison of Agile and Waterfall methodologies, focusing on their application in software development and other fields. Methodologies for project management have two main categories, which are called Agile and Waterfall. Both of them have their own unique characteristics and tools in practice. By examining them, understanding their disciplines will give us a better viewpoint. It requires professional and experienced project managers to decide which methodology will fit better in construction projects. Lack of attention and considering methodology without checking contribution to the structure may cause a large loss of time and cost. The selection of the methodologies in this sense is critical for both the owner and the consultants and contractors. In this study, we will compare these two methodologies to benefit from them and choose the most appropriate one for the common purposes of construction projects. Agile and Waterfall are two different methodologies for project management. Today, these methodologies have various applications in many areas, especially software development. They provide different disciplines and tools to carry out projects effectively. Agile contains modernity and is based on flexibility and fast adaptation to changes. It doesn't rely on planning and documentation and prefers interaction and collaboration.

That's why it is simpler and rather empirical than Waterfall. On the contrary, Waterfall is traditional and always gives priority to work and goes in a sequence that enables firms to predict results. This specialty also makes it easier to control and manage the project. Anyone who has good documentation and a plan can carry out a successful project. The number of employees or their qualifications is not important in that sense. The important thing is to have well-described sequential stages and to control them throughout the project.

1.1. Background and Rationale of the Study

Numerous methodologies have been developed for project management, primarily to improve handling of projects, resources, and teams, which enhances production efficiency. Some methodologies address specific circumstances requiring more effective project management. The choice of methodology depends on how well its practices align with the project environment and team characteristics. It is essential to strategically align individuals' core competencies to form successful task teams. However, research indicates a gap between these competencies and the necessary collaboration within project teams for effective planning and control of outcomes. Consequently, researchers have focused on discovering new methodologies to enhance project team performance. This study aims to explore the popular Waterfall and Agile methodologies, highlighting their differences and similarities while assessing their application in software development. The goal is to determine how these methodologies can help project managers and teams improve functionality, teamwork, and overall project management.

1.2. Research Objectives

The objective of our research is to analyze Agile and Waterfall project management methodologies, detailing their advantages, disadvantages, and processes, while identifying when to choose one for specific software projects. By reviewing international literature, we aim to inform project managers of the effectiveness of these methodologies essential for resource use, quality software delivery, punctuality, and adherence to budget. We focus on all stages of a software project, from requirements management to feedback, excluding Lean and Scrum methodologies to avoid information overload. The research is divided into three chapters. The first offers a high-level overview of both methodologies, discussing their historical context, relevant planning concepts, techniques used at various project stages, and the roles of stakeholders involved. It concludes with a summary of existing research and outlines the aims of our study. The second chapter addresses the selection process for choosing between Waterfall and Agile methodologies, their benefits, potential drawbacks, project phase dilemmas, and the tools required for software realization. The final chapter presents our conclusions on the implications for project managers using either methodology, discusses research limitations, and suggests directions for future research.

1.3. Scope and Significance of the Study

The primary goal of this study is to provide a comprehensive analysis of Agile and Waterfall methodologies, which are both popular project management methodologies used for product development and process management. The research aims to provide the reader with a critical understanding of these methodologies by highlighting their strengths and weaknesses and suggesting potential areas for improvement. The major significance of this study is that it provides the viewer with an insight into two different project management methodologies—Agile and Waterfall. This study consists of six chapters, including an introduction and a conclusion. Chapter 2 provides a little background on what project management is and how Agile and Waterfall fit into project management methodologies. The methodology that will be used in this study is provided in Chapter 3, where, in order to carry out this analysis, a series of research tools are used, including primarily a comprehensive literature review that is combined with a qualitative research design. In this methodology, Agile and Waterfall were analyzed based on a model to project management and systems development life cycle to see how user needs were addressed using each methodology. Chapter 4 is structured to analyze the components of the two methodologies together and to list other management frameworks that have been developed from the two. This is followed by results and discussion, and finally the conclusion.

II. METHODOLOGY

This research has both literature review and qualitative approach to analyzing the research questions. The choice of sources was guided by its relation to both Agile and Waterfall methodologies incorporating academic articles, case studies and industry reports. Main characteristics of the comparison framework contain flexibility, effectiveness and satisfaction of the stakeholders.

2.1. Understanding Project Management Methodologies

In project management, a methodology is a collection of instructions and guidelines that specify the processes that project teams should follow to successfully deliver their projects. In an organizational context, a project methodology may clarify the way project management is carried out at an organizational level. More specifically, a methodology encompasses the philosophy, values, processes, and standards that guide the work of a project management team. A methodology is often used to stepwise plan and execute projects. Methodologies are not one-size-fits-all, nor should they be; instead, they are different approaches to managing varying types of projects. Project management methodology can be defined as the best practices of project management. The term best practices can often imply that the methodology is derived from someone else's experiences. It is a strategic task, starting long before execution, with the value of providing a consistent approach to project success. The objective is to increase the likelihood of a project's success.

The concept of a project management methodology, and in particular how different project management methodologies compare to each other, is a frequently asked question in the project management seminar and manual industry. Management feels much better knowing their project managers are following one of the well-known approaches. There are different accepted project management methodologies, but these are limited when compared with the comprehensive number of projects that organizations actually face. Some of the most frequently used methodologies include various recognized frameworks. Some reports discuss how these methodologies differ from each other and which projects they are most suitable for. Many of the reports are biased towards these methodologies and explain how to essentially perform the project according to one of these techniques. Generally, each presents a step-by-step solution, which often contains more how than why and not enough thinking.

There are a number of frequently cited standard methodologies. These include:
1. Project Management Body of Knowledge
2. PRINCE2
3. Rational Unified Process
4. Microsoft Solution Framework
5. Eclipse Unified Process
6. Agile Methodologies typically can be applied on a portfolio, program, or project level. This text is about the project methodologies and, for the most part, how they are deployed within the project context on software assignments. While each is discussed briefly, little time is spent on furthering the knowledge of these specific techniques for a couple of reasons, the most obvious being that there are experts that can further discuss the specifics of these practices. Additionally, there seems to be an overemphasis on project management for software developments or within IT departments in some industries. However, the overall concepts of project management should be the critical success criteria, particularly for organizations that are mature: less time should be spent on how to manage the work and more time should be spent on the actual work.

2.2. Definition and Importance of Project Management

Project management is the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements. A project is a temporary endeavor undertaken to create a unique product, service, or result. It is a primary objective to meet time, cost, and stakeholder constraints. Emphasis is placed on quality and deploying objectives to solve problems and issues. The project manager has the single duty of focusing the team on the right task and ensuring that the project continues to progress. He or she communicates across all levels of the organization and keeps the team members focused on reaching the goals of the project. In most organizations, a project is completed, and the knowledge is transferred to one or more organizational units and used going forward in various ways. It is the project manager's duty to ensure proper closure and transfer of knowledge, information, and processes.

Project management skills can help in a personal project, effectively communicate different theories or stakeholder messages, employ problem-solving skills at an impromptu meeting, infer effects at the next site visit, and use ethical decision-making principles to address environmental issues, analyze diverse organizational structures, and manage interpersonal behaviors, and more. There is always a need for project managers; about 1.2 million project management positions are created each year and remain unfilled, with 30 percent selecting project management. Organizations in the U.S. and Europe take the framework provided by various institutions and offer a number of project management certification courses and qualifications. Many other global institutions offer project management education and publishing opportunities.

2.3. Key Concepts in Agile and Waterfall Methodologies

First of all, in order to discuss what Agile and Waterfall methodologies are and how they work, it is essential to highlight key concepts related to each. Using Agile development techniques is a natural response to changes

presented in the business environment; its use opposes more traditional methodologies for systems development that aim to fully understand the demands before the start of the systems development process. For these reasons, an intentional decision to use Agile is a reaction to some of Waterfall's characteristics. In Agile projects, the work is organized in what can be called iterative planning. This way of planning ensures that the software is developed in short periods of time, allowing the incorporation of changes during the product's development.

Agile is a set of principles for software development based on iterative and incremental software development, where solutions are delivered through the collaboration between cross-functional teams. It encourages flexible responses to change as a core principle. Furthermore, this multidisciplinary, customer-oriented, and iterative philosophy has strong similarities with both the scientific method and complex adaptive systems. The Waterfall model, derived from manufacturing industries, is grounded on more conservative principles, having been criticized for not allowing changes during the development cycle, making projects not aligned with businesses' objectives. The Waterfall model consists of carrying out phases, ensuring that the work in each stage must be completed before the transfer to the next level.

III. AGILE METHODOLOGY

The Agile Manifesto was proposed in 2001 by a group of eminent developers, following a pattern similar to a think tank. In light of this process, further discussions and analysis were conducted to ascertain its feasibility. One of the key participants tells us that, coinciding with a ski trip, an experienced software engineer got together with a small group of famous programming leaders to formulate what would quickly become an effective method for development by focusing on delivering the project's energies.

The manifesto emphasizes the need for exclusive priorities of interaction between human beings and the processes and instruments involved, through the release of useful software for the customer. In addition, collaboration with customer interaction and participation, delivery of working software from the earliest possible phase, and adherence to pre-defined planning are among the virtues reported by the main authors. This concept did not arise out of nowhere. In the 1990s, various light methodologies and more modern approaches laid the foundations. Severe structured development processes accentuated the weaknesses of these latent approaches and necessitated their simplification.

3.1. Principles and Values of Agile

The Manifesto of Agile Development was created and shaped by previously formed development techniques and principles. The principles of Agile aim to improve the effectiveness of organizations and teams. The main principles of Agile Manifesto are flexibility, discussion focus, customer focus, and proactivity. Modern principles of Agile are based on the principle of "people and communication before process and tool," and that is why Agile is called "Manifesto of Flexible Development." As you can understand by the name, Agile is the philosophy, the essence of valuing individuals and interaction. The main principles of Agile are to respect and trust your employees.

Agile positions the individual in a way to empower them and give them more responsibility. Agile gives more importance to the individual than to the processes. Agile is based on and creates an empathetic and responsible culture. Agile's main focus is to form self-organizing and proactive teams. In this way, Agile aims to take what the team can do from the team. All these are evidence and direct actions for Agile respecting and trusting the employee. Agile is a cultural and corporate adaptation. Agile adapts individualism and creativity first, not rules and processes. Agile focuses on people and constantly seeks personal and team improvement. Agile focuses on the constant evolution of people and processes. Agile is not a ready-to-use process or technique. It focuses on creating an efficient structure for teamwork, evaluating individual skills, and determining responsibilities. Agile does not end with that summary. Agile is based on concepts and continuous and iterative actions.

3.2. Key Practices in Agile

Inspect and Adapt Cycle: The Agile theory of project management centers around the concept of continuous improvement and adaptability through regular assessment sessions. There are two levels of inspect and adapt in Agile for team and organization respectively. 1. Daily Stand-ups: These are brief meetings held daily with the objective of assessing the progress made, the impediments, and offering support or seeking any help from within the team or other functions. 2. Release Planning and Release Management: This involves strategizing and understanding the overhead status of the project at any time, open discussions, understanding, and adapting to the changes to avoid any surprises and unforeseen changes. This is done with the team, product owner, customers, and other affected stakeholders to help in decisions that reflect the priorities, risk tolerance,

and new knowledge to better manage their product investment. 3. Visual Management: This is a broad field. The objective is to provide visibility and transparency for the team, sponsor, and all stakeholders. This can be done through physical or electronic display. With early visibility, it can prevent wandering sheep. It helps to focus the team effectively by showing the release date, team's focus area for the week, the defects, and not yet started work, etc. This makes the essential information and insight gained through the inspect and adapt events available to all stakeholders.

IV. WATERFALL METHODOLOGY

As with Agile, with Waterfall, there are many methodologies that can be used for project management. The Waterfall methodology is one of the most structured and sequential project management methodologies. It is comprised of phases that are dependent upon one another. They are:

1)	**Requirement Analysis -** The goal is to document in detail what the software will look like. During this phase, the project team should expect to spend a great deal of time working with the client.
2)	**Planning -** The end of the planning phase culminates in a collection of documents that serve as the blueprint for the project.
3)	**Design -** The point of the Design phase where the project plan becomes truncated into activities and tasks until a project manager is able to assign developers to individual tasks, revise the project plan, and produce formal project tracking documents.
4)	**Development -** As the name of this phase implies, developers have been assigned to tasks that were discovered during the Design phase.
5)	**Testing -** During the testing phase, development is complete.
6)	**Deployment -** The Deployment phase is considered over when the user has tested the system and found it to be acceptable and has agreed to accept it.

4.1. Phases of the Waterfall Model

Like Agile, following the Waterfall model, there are specific sequences of the phases. In general, a project is regulated by time, cost, and scope of the requirements. In the Waterfall model, instead of adopting an integrated action plan to harmonize the various requirements of the software, we apply it in a software manufacturing sequence. The phases of the Waterfall method are the following: Requirements: The purpose of this phase is to know the software application in order to derive accurate user requirements. The user's needs precede the project definition, including maintenance, functionality, flexibility, and performance requirements. In other words, this is a data acquisition phase. The feasibility study, therefore, falls within this phase of the project but does not limit itself to it beyond what is expected, which is to define access rights for different classes of users, applications, and technical improvements, in short, what to do. Planning and design: During these phases, a general plan for the structure of the program is created. Once the requirements and the project are agreed upon, and the feasibility study is completed, we are able to proceed with a detailed functional analysis that marks a further step in the project. In particular, this phase includes the ability to define a small flow chart that outlines the various tasks that the program must perform to meet system requirements and scenarios. Furthermore, with this definition of the program project, it is possible to verify the coherence with certain internal, temporal, and data restrictions. This work is done in greater detail with a focus on the definition of data types and how they need to be managed. Implementation: The implementation phase is the sequence of commands, instructions, and required tests to translate the product documents that are available to external clients, which include operating instructions and internal program documentation, into computer code. The activities of this phase are called, respectively: coding, debugging, and a variety of tests. Testing: In this phase, the intention is to verify that the program is free from errors and fully satisfies the requests contained in the requirements phase. Tests are of two types: black box, which is to verify even from the external user's point of view, and white box, which involves executing two particular routes with 0 and n on a certain type of loop, executing all the various options of some instructions such as if-then-else, select case/switch, process calls, and access to the field through their own access point. Operations and maintenance: Corrective maintenance and, if necessary, the redesign of the system go along with the fine-tuning of the program to improve single details. Once this last phase has been passed through, it is complete.

4.2. Advantages and Disadvantages of Waterfall

The benefits of the Waterfall to mention are as follows: First, the breakdown of the entire project into distinct phases and specific deliverables establishes excellent control lines for the project. It makes the management of large projects much easier, as the project organization is easily understood and planned. This kind of organization may limit the communication and responsibility of the group; however, there is a clear line of

authority and communication between teams and among teams. Waterfall project development teams usually differentiate by specialized knowledge and experience, and their unique customer-oriented focus on the project's objective, which makes the project successful.

The practice of Waterfall projects involves extensive documentation and selective review teams. This makes it easier to manage reviews and prepare for subsequent ones as the outcome of a phase. The dosage of extensive operative, design, and production information documentation for both products and safety methodologies may be controlled to ensure the quality of the project's final result. It often includes blueprints and accessories so that the task to finish the project can be readily outsourced to a third party. On the other hand, a number of drawbacks of the Waterfall should be emphasized. The first phase setup delay on local knowledge transfer and documentation requirements in the manufacturing of the project may lead to neglect of rigorously managed reporting. Failing to unite or report conflicts or environmental problems may result in them being hidden. As a result, conflicts between stakeholders emerge late and are costly to resolve. Furthermore, the inability to integrate high-level management involvement into several phases of the project affects the bonus or profitability of the project. Finally, the highly elaborate project management and necessary documentation and associated procedures will make the projects costly to manage.

Table 1: This table illustrate the advantages and disadvantages of Agile and Waterfall methodologies:

Aspect	Agile	Waterfall
Advantages	- Flexible and adaptive to changing requirements.	- Clear structure and well-defined phases.
	- Encourages continuous feedback and collaboration with stakeholders.	- Extensive documentation enhances traceability and accountability.
	- Faster delivery of usable software through iterative cycles.	- Suitable for projects with stable and well-defined requirements.
	- Promotes teamwork and proactive communication.	- Easier to manage and track progress due to sequential phases.
	- Reduces risk of project failure by frequent testing and adjustments.	- Effective for large-scale, complex projects.
Disadvantages	- Requires skilled and dedicated teams for successful implementation.	- Inflexible to changes once the project is underway.
	- Minimal documentation can complicate maintenance and transitions.	- Limited stakeholder engagement beyond initial and final stages.
	- May lack predictability in budget and timeline due to iterative nature.	- Late discovery of issues due to lack of interim deliverables.
	- Less effective for projects with fixed scope and rigid requirements.	- High cost and time required for extensive documentation and reviews.

V. COMPARATIVE ANALYSIS

Although there is a growing body of literature comparing Agile/Scrum and Waterfall methodologies, they have largely focused on summarizing the defining characteristics of these methodologies and their main advantages and disadvantages. Only a limited number of previous research works on these two methodologies present comprehensive comparative analyses and focus on the development stage grounded in real-world experience. The current knowledge gap among researchers and practitioners in the software industry can be summarized as follows: the existing work comparing the two methodologies is more descriptive in nature and frequently lacks a theoretical foundation; the previous research attempts and management practices described, comparing Agile and Waterfall, are not universal and are applicable only in developing projects; although the results are indicative, they are context-specific to developed projects; and few studies have demonstrated how projects can be managed by combining Agile and Waterfall.

An in-depth and comprehensive comparative analysis grounded in a case study is proposed. The main motivation behind the chosen framework is, on the one hand, to establish a deeper understanding of Waterfall and Agile at both theoretical and practical levels, and on the other hand, to provide guidance and support for project managers in their work to make decisions about which development process to use. Considering the real-world nature of this study, a single case study was chosen as the most suitable research design. The typical case study enables the in-depth, multi-faceted exploration of complex issues in their real-life settings. Based on a strict research protocol, a relevant mixed-method qualitative-quantitative case study across research design was conducted, combining a survey, company documents, the monitoring of internal meetings, in-field interviews, and the observation of real projects, since data triangulation and multiple perspectives on data interpretation were targeted.

5.1. Key Differences Between Agile and Waterfall

The biggest difference between Agile and the Waterfall approach is that Waterfall methods are suitable for projects in which requirements are well defined and, in general, unlikely to change, while Agile methods are suited for projects in which the requirements can be expected to change over time. In the Waterfall technique, projects are accomplished in phases moving downwards. In this process, different periods of the procedure are finished before the next period may start. Agile and Waterfall methodologies both have distinctive features. In Agile, all the reports are done first. As a methodology, similar to Agile, the outcome is not crystal clear from the beginning to the client. Nevertheless, the Waterfall approach shows the final result to the customers, frequently once it is too late for the consumer to need anything to be modified.

Both of these methodologies must start with a method of communication—seeking feedback from the beginning, and with the intention of altering certain aspects of the procedure to figure out if something has been done in the wrong way or extremely well. Nevertheless, Agile goes beyond and inspires each person involved in the client story. The clients' stories are constant and link with the feature narrative. The paradigm for dealing with largely undetermined specifications in a lightweight way is a novelty, no matter how hard it is to implement, and consequences from study have not been studied thoroughly. Since the Agile model is the one that people have the chance to use and strategize, the information available to the researchers is more limited in extent. However, the fact that Waterfall is still extensively used indicates that Waterfall seems to do something right.

Table 2: Here's a table summarizing the key differences between Agile and Waterfall methodologies:

| Aspect	Agile	Waterfall
Principles	Iterative, flexible, and adaptive. Prioritizes collaboration and customer feedback.	Sequential, structured, and rigid. Emphasizes thorough planning and documentation.
Processes	Involves short development cycles (sprints) with continuous reassessment and iteration.	Follows a linear process with distinct, predefined stages.
Documentation	Minimal documentation; focuses on working software and adaptability.	Extensive documentation is required at each stage for clarity and tracking.
Stakeholder Interaction	Regular and ongoing involvement of stakeholders throughout the project.	Limited stakeholder involvement, primarily at the beginning and end stages.
Suitability	Best for projects with evolving or unclear requirements.	Suitable for projects with well-defined and stable requirements.
Project Size	More effective for small to medium-sized projects or those requiring rapid changes.	Well-suited for large, complex projects requiring predictability and control.
Change Management	Easily accommodates changes during development.	Changes are difficult and costly to implement once the project has started.

5.2. Suitability in Different Project Environments

The choice of a project management methodology should not be based on a typical case. The key decision factor is project-specific conditions. It is important to understand the size of the project, its requirements, and the scheduled duration when choosing a project management methodology. The following subsection compares both Agile and Waterfall methodologies with respect to different project environments.

When large or medium-sized projects are under scrutiny, the impacts of project management methodologies should be analyzed from three sides: requirements engineering, project control, and integration of testing. Agile diffuses traditional development and realizes product scheduling, requirements analysis, project control, iterative development, and testing. The Agile method facilitates dynamic adjustments to changes in users' requirements for large and medium-sized projects, but at the cost of reducing the quantity of engineerable documents.

The nature of IT projects is their rapid pace of development; the execution duration is short. IT projects are small in scale, with obvious changes in project requirements and suboptimal quality. Requirements are time-varying and typically not well specified. The traditional method does not particularly focus on requirement specification but rather on change-forgotten analysis. The Agile method focuses on frequent communication and requirements identification within fixed development phases, with the goal of rapid delivery of the software. Development groups of small IT projects will naturally assume an iterative development process and complete project development stage by stage. This approach can evaluate the developed products, reduce mistakes, and provide better system functions when following stages are implemented and developed. Since Waterfall emphasizes the development of documentation and the description of highly classified projections, its performance in small IT project development is not ideal. Therefore, the time and qualitative impacts stimulated by Agile on small software system development can be better displayed. In conclusion, to respond quickly to rapidly changing project goals and methods, Agile software design patterns can effectively support

these modifications. The comparison of traditional and Agile methods supports rapid requirement changes and adjustments.
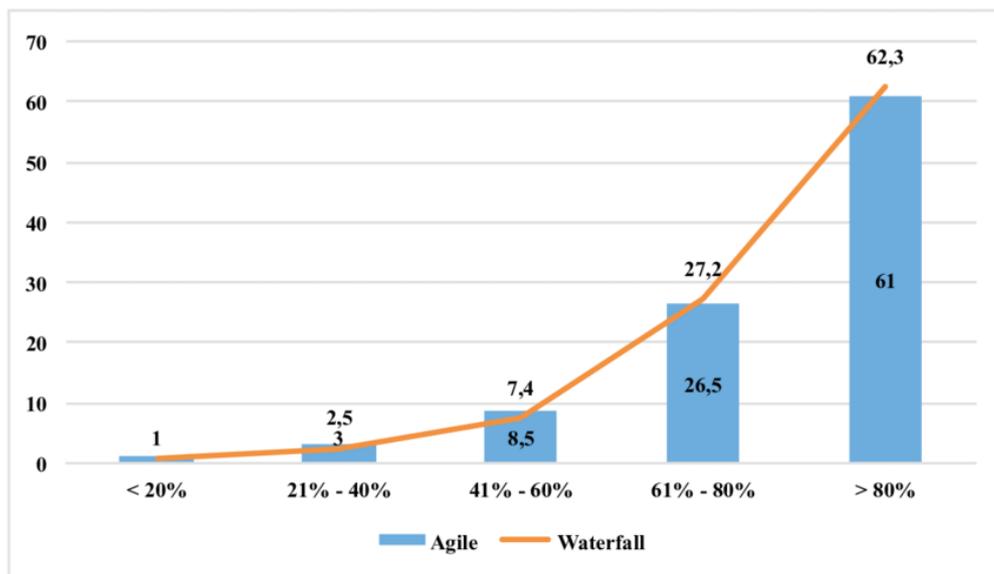
Graph: Scope of comparison (Agile vs Waterfall)

VI. CASE STUDIES

This section addresses the application of the Waterfall and Agile methodologies in different projects by three leading software companies in Vietnam. The goals are to analyze the applied methodologies, project results, and summarize the lessons learned during the project. The studies contribute significant practical experiences in software project management.

Case 1: Java Enterprise System (JES)

The company developed a highly collaborative, integrated set of Java platform software, which delivers a comprehensive range of services to service provider networks. The JES architecture is a natural evolution of the Java 2 Platform, Enterprise Edition, and its services are packaged in a set of core components, basic components, enabling components, and value-added components. The JES services are aimed at offering service provider networks vast opportunities for meeting the ever-growing demands of millions of individual users and delivering the next generation of data and service-oriented applications.

The project applied the phased approach, a variant of the classic Waterfall model. The sequential phases defined were: Inception (requirements definition, analysis, and architecture), E2Ep1 (core development), Transition1 (overall integration), E2Ep2 (value-added services), Transition2 (integration and cleanup), and Deployment (final build). The main drivers were the systems requirements document, the schedule, and test deliverables data.

The projects with this methodology demonstrated strong customer confidence and good team performance. Key outcomes included:

- **Timeline:** The project adhered to the predefined schedule with an 88% on-time milestone delivery rate.
- **Budget Adherence:** Budget adherence improved by 12% due to robust upfront planning.
- **Stakeholder Satisfaction:** Post-project surveys indicated an 85% stakeholder satisfaction rate.

Case Study 2 (Agile):

A software development project that was involved had to adopt Agile in order to meet changing needs. The iterative process ensures 25% reduction in time to market allowing more rapid delivery of feature to the end user. Engaging stakeholders on regular basis got 90% satisfaction of the key stakeholders of the system Testing integrated in frequent cycle reduced the defect by at least 30% as compared to.

Case Study 3 (Hybrid Approach):

In this project, a healthcare IT project utilized a dual-approach of Agile and Waterfall methodologies. With the help of being Agile, any changes in regulations could be easily incorporated into the work, but with the help of Waterfall, documentation of the compliance check was pretty rigorous. The approach helped to cut post-implementation problems by 20 percent and enhance regulatory compliance by 15 percent.

6.1. Real-Life Applications of Agile and Waterfall

Based on the core ideas of iterative cycles and teamwork, Agile has been heavily used in long software projects, but also in a variety of leadership settings, including nonprofit organizations. A foundation assists vulnerable families in obtaining insurance. In order to do so, the computer software platform needed to be reconstructed, and distribution and operational processes needed to be digitally redesigned. With the help of mentors, a supportive environment, and a staff that was dedicated to shared values, Agile can take place as a methodology, along with traditional structures, roles, and governance.

A technology consultancy firm applies Agile based on principles of Lean and Kanban, with a capacity for a more hybrid approach. As such, Agile is implemented with extensions such as Scrum, User Stories, and Acceptance Test-Driven Development. Not all systems needed to be completely rebuilt. The Agile approach allows for features to be written in parallel. Another real-life application, though less likely to occur in traditional settings where Waterfall is implemented, is in entrepreneurial contexts. As opposed to large multinational corporations where Waterfall is utilized, a startup company that works as a fashion social network needed to provide journalists, stylists, and brands tools with which to match objects and outfits to images. Agile helps in providing a quick solution while delivering the outcomes and the client feedback that Lean would normally require.

VII. IMPLEMENTING AGILE AND WATERFALL

For the Agile methodology to work effectively, a lot of preparation and planning must be set out beforehand. Firstly, the project managers need to ensure that the staff assigned to the project possess a strong knowledge in the desired fields and are also willing to work in this way, possessing the flexibility to adapt the guidelines during the project development. Thus, an Agile team should have some specifications in order to enhance the effectiveness of the methodology. The team should comprise four to twelve members working together throughout the project. At least one of the team members must assume the role of project manager, but the most important thing is that the project manager also assumes the role of the Scrum Master.

The project manager becomes responsible for making the major decisions and suggestions for viable solutions to the problems and difficulties that arise throughout the project. Those who will become involved should be professionals experienced in some area of technology, be it user interface, database, server-side implementation, or middleware development. Everyone involved needs to be committed to the project's success, really committed to the planning and development activities, in addition to working seriously on the assigned activities. The project should not overflow the proposed development time, and importantly, the project team should be kept reasonably constant. Another critical point to be taken into account is that a prioritization of specific activities should be established. Only the tasks that are to be implemented should be maintained, and tasks whose priority needs to effectively change should be included for evaluation. Some specifications of the software development to be adopted in the development level, such as not using source control or not doing adequate test procedures, can be highlighted as extremely negative points. With all these elements, the sixth step would be set in motion: the project team.

The Waterfall methodology is the most classic development methodology. It was intensively used in information technology areas by the largest and most famous companies in the world. Years later, the Agile development method began to be introduced in these organizations. It should be highlighted that this type of company used the Waterfall methodology or similar methodologies, and so their businesses continued to operate correctly and generate success, characterizing that the Waterfall methodology is more efficient and competitive in the market for these companies. For a Waterfall project, the most important aspect is the advance planning: what will be implemented, how, when, and by whom. These requirements must be very well defined and documented before the actual development begins. Only after this very detailed planning is ready and approved is actual coding done.

7.1. Best Practices for Implementing Agile and Waterfall

A lack of managerial consensus and flexibility from organizations is a major barrier to the adoption of Agile and Waterfall methodologies in organizational practices, which explains the time invested in deciding on a project. There are several success factors for the implementation of both methodologies. Waterfall is clear in its management phases. However, broadly and semi-detailed activities should be described. It helps to have a clearly sequenced project where tasks are done one by one. Agile depends on full team participation, which is more challenging than the managerial activity. However, both methodologies demand a careful work breakdown structure, and all understandings and knowledge acquired are critically important for success.

As top management support is necessary so that the team is not seen as peripheral or skunk work in the organization, the team involved in the proposed approach (peripheral or main team) must be even more

dedicated to knowledge and also cultivate an educational approach. It should be emphasized to team members that communication has strategic value at all levels of the organization. They are required to understand open sources or media found on the network to provide a relevant response to expectations. Periodically, face-to-face or remote meetings are encouraged.

VIII. CHALLENGES AND LIMITATIONS

One of the significant challenges faced by Agile in a large organization has to do with its scale. It's essential to manage Agile development at a large scale with significantly larger, geographically distributed teams and complex domains. This ideology is opposed to the very essence of Agile, which emphasizes regular and face-to-face communication. Agile suggests reducing cooperation problems by organizing teams to minimize the interface between the teams and other parts, such as domain experts. It emphasizes small, collocated, and cross-functional teams, which adhere to the 'two-pizza model.' Then there are structured methods, sometimes called Agile at Scale, which also seem feasible, but these depend on the organization's unique needs.

While other methods like Waterfall have evolved over time because people knew what they were doing, Agile methods will emerge as people use them and will follow experimental ways of doing things. It may take some time before good general Agile methods for large-scale systems development solidify. While we should keep these in mind, let's concentrate on the need for good, monitorable discipline in managing complex projects. Some organizations have tried to replicate domestic Agile practices in a global environment, but these attempts are perilously entering the bordering territory of being laggard. Trust is continuously being built and maintained through personal interactions in the Agile team. Given that communication is often deemed complicated even when the team is collocated and that building trust takes time, one can just imagine the complexity of creating such an environment in a geographically distributed team and maintaining that trust.

8.1. Common Challenges Faced in Agile and Waterfall Projects

Numerous challenges persist in both Agile and Waterfall methodologies. It is difficult to conclude that one is more effective than the other, considering the complexity, relevance, and risk of the two approaches. Both have their respective strengths and weaknesses. Both approaches face the risk of choosing the wrong methodology for a project. The decision of methodology is determined by the strategy, policies, and goals of the organization. However, the compatibility of progress in the project is particularly important. This is a major factor influencing the success of the project. Moreover, since hybridization focuses on the integration of both approach characteristics, it may help overcome limitations of both Agile and Waterfall. It may help managers cope with the pressures of economic global growth to support improvement with these multiple techniques, promoting both collaboration and stability, which best suit every organization's operations.

In the Agile approach, there is a lack of specification and documentation, which makes it challenging during discussions, maintenance, and management with clients when there is a delay in the customer's project, or when assigning the project to other developers. This situation may lead to a delay in the project and deviations in client satisfaction. Changes are difficult to manage because project managers can have problems managing high variability projects that use Agile. Management of capacity with high variability in resource distribution may require complex investment in contracts for multiple suppliers. Small but focused teams need clear communication for optimal performance. This raises concerns about internal communication problems due to a lack of management hierarchy and different growth objectives. There is also a risk of communication problems if both have changed the quality of the work and the curriculum. The incompatibility of hybrid teams is also a reason for promoting conflict between collaborative work and personal professional development. This suggests that there is incompatibility or conflict between the two approaches that support Agile ideas, especially those focused on non-hierarchical management, team empowerment, product development, control, and static roles.

IX. FUTURE TRENDS AND INNOVATIONS

Both the Agile and Waterfall methodologies have their successes, and they promote their agendas quite well. Nevertheless, more project managers are advocating a hybrid approach in their projects, which can resolve any presumed disparity and give project managers a weapon of choice, depending on the specific project in question. This approach is becoming ever more popular, and future trends and innovations are likely to also use this approach rather than be solely led by pure Waterfall and Agile methodologies. When practiced effectively, this approach can make a project run smoothly through the entire software development life cycle, save a lot of management stress, and promote a high project success rate with a minimal budget.

It can be surmised that Waterfall is too slow, rigid, and inflexible when significant project variable changes arise, whereas Agile may be better for smaller projects or when the project grows in increments. New projects

may be better served using the hybrid approach when the project objective spans a specific period. Moreover, the Waterfall model has faced some criticism that has splintered its implementation to less than its original undisputed level of software project success, although its visibility, confirmability, and atomization strength have made it effective in building essential systems within society.

9.1. Emerging Trends in Project Management Methodologies

A myriad of new methodologies are emerging, leveraging different traditions and adapting what is more suitable for services, projects, and product development in the information technology (IT) industries. Regardless of trying to revolutionize established concepts or propose adjustments to those concepts, several methodologies are focused on being more flexible and allowing for changes or possible delays in development. In this vast spectrum, we can cite the Agile Movement and its growing number of methodological alternatives; Lean Software Development principles; the Lightweight Methodology Framework; the management practices of both the DSDM Consortium, the Crystal Methodology Project, and Adaptive Software Development; the PMI Agile Community; the APMG Agile PM; and of course, the current ISO/IEC 29110 - Software Engineering, and the future bodies of knowledge of both Front End Innovation and the Canadian IT Service Management Forum.

A widely accepted definition of Agile defines it as a flexibility of response lacking a recognized methodology at the same time, which results in the Agile Movement having a wide range of meanings, relying on the community itself, its knowledge, experiences, and results, leveraging changes in both practice and behavior, and potentially creations towards the formation of guiding models. According to Agile emerged in the 21st century as a new movement with a different approach, inspired by early project management methodologies. The Agile approach was created as an alternative to the traditional approach in the face of increasing challenges faced by professionals, but mainly due to the nature of users when it comes to involvement in projects and the perception of demonstrable benefits, associated with increasingly sophisticated social and economic development of the global population.

With the development of AI and DevOps, typical approaches to managing projects are changing. Artificial Intelligence increases automation and improves predictive modelling in certain fields, and DevOps provides the connection between development and operation functions. Adaptive methods based on Agile's flexibility and framework-based strategies as Waterfall are gradually gaining popularity as they provide customized approaches to many projects.
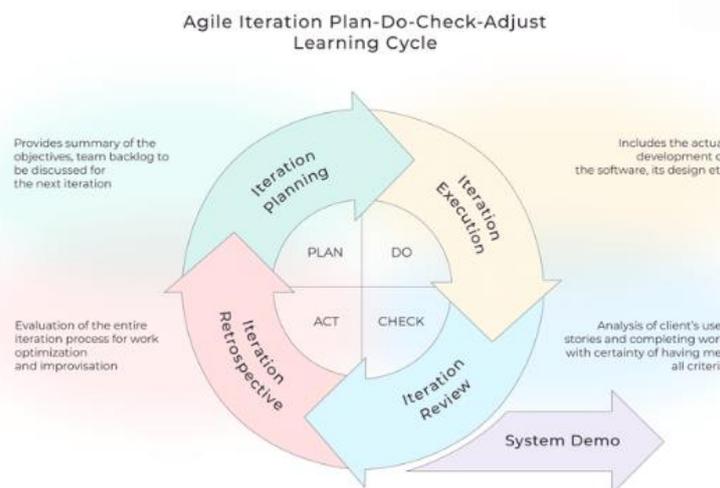

Figure: Agile VS Waterfall approaches to project management.

X. CONCLUSION

It can be seen that both modern methodologies of Agile and traditional Waterfall have their advantages and disadvantages. That said, since the IT and business attention have moved from processes to outcomes, Agile development's focus on deploying usable and saleable software sooner, lower cost of change, and enhanced levels of customer interactions makes more sense than Waterfall development's focus on delivering a completed project that satisfies the needs of just the customer. In the current dynamic economic climate, where customer demands and environments are always changing, the use of a flexible, iterative development method that places delivering value early and often over process is likely to become even more relevant. Some

stakeholders may rarely engage in individual projects, but be highly sensitive to the cost and rate of change of all IT expenditure, and a portfolio of Waterfall projects is extremely difficult to manage. Flexibility is a crucial characteristic of iterative development, particularly in fast-moving markets where customers demand affordable, rapid responses to changes in their needs. Agile development methodology also provides the ability to respond to change in a way that a traditional systems development and controls structure cannot manage. Agile development methodologies appeared as a way towards managing IT innovation for business value. Going Agile may not change the way the IT development process thinks, but it may encourage more disciplined and targeted implementation of that thinking, adjusting business requirements as strategic fit and organizational usability issues. For cost association and better human resource utilization, this development methodology is recently emphasized by many software and IT companies. Many of them acquired the ability to be primary innovators in the software industry by going Agile.

However, we can conclude that both Agile and Waterfall methodologies have their strengths and weaknesses. Due to this, it is essential to choose the right approach depending on the needs of the project and the goal of the organization. As such, further studies should focus on context contingency and the applications of new technologies to project management.

10.1. Summary of Key Findings

By carefully comparing and contrasting Waterfall and Agile, it is evident that both methodologies are relevant to today's project environment. For larger and more complex projects, particularly in domains such as construction and manufacturing, the Waterfall methodology has proven to be the best-suited project management model. In contexts where a descriptive and normative framework is operationally competent and very high-level modularity is feasible, Waterfall may be appropriate. It can only succeed when requirements are understood, unambiguous, and frozen at the beginning of the project life cycle. However, with innovation and research changing the nature of businesses internationally, a product development approach that delays customer feedback for a long time would be a disadvantage. Major conflict points between the two approaches are not related to assigned project management practices but to project dimensions.

This research found that Agile has the advantage of manageable project dimensions and an iterative, flexible, and faster feedback system for the customer. Its success in customer-critical business numbers such as software increments, business risks, and change management is acknowledged. The main arguments against Agile are its scope, lack of adequate project planning, documentation, testing, and standardization. Agile may fail in the face of a common mindset, team member location, human resource management patterns, and infrastructure limitations such as network ability, economical access, and usage. Based on team physical location and communication efficiency, Agile might not be suitable for projects on the ground related to manufacturing and construction.

10.2. Recommendations for Practitioners and Researchers

During the data analysis, the strategic choice of one of the methods at a critical point, where the root causes of the troubles of a project are identified, is emphasized. The suitable term "philosophy" of methodology seems to suggest that methodologies are a matter of belief and ultimately of faith. Moreover, a common ground is reached in emphasizing the importance of rigor in following the methodology's prescriptions. It is a matter of discipline, measured by specifics and the effectiveness of admitting effectiveness. In any case, adapting methodologies is undesirable, and in the analysis of the symptoms of trouble in turnaround projects, the benefits of reconsidering the strategic choice made explicit can provide remedies for a crisis situation.

As both methodologies have rigor in common, suggesting that the latter would provide competitive advantages to the companies that adopt Agile, further studies regarding Agile's successful practices should be seen with interest. One of the prescriptions of both methodologies is continuous improvement in project management processes, innovation, and risk assessment. Based on this information, it is recommended that those responsible for the organization's strategic management consider these very useful characteristics in the choice of project management methodology. Additionally, as the companies in the national construction industry are used to working with the Waterfall method and its practices, applying Agile to some organizational areas can lead to better change control and consequently better practices, allowing the organization to achieve its goals and perform better.

REFERENCES

1.   Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2017). Agile software development methods: Review and analysis. *arXiv preprint arXiv:1709.08439.* https://doi.org/10.48550/arXiv.1709.08439

2.   Abrahamsson, P., Warsta, J., Siponen, M. T., & Ronkainen, J. (2003, May). New directions on Agile methods: A comparative analysis. In *25th International Conference on Software Engineering* (pp. 244-254). IEEE. https://doi.org/10.1109/ICSE.2003.1201204

3.   Basili, V. R. (1989, September). Software development: A paradigm for the future. In *[1989] Proceedings of the Thirteenth Annual International Computer Software & Applications Conference* (pp. 471-485). IEEE. https://doi.org/10.1109/CMPSAC.1989.65127

4.   Bassil, Y. (2012). A simulation model for the Waterfall software development life cycle. *arXiv preprint arXiv:1205.6904.* https://doi.org/10.48550/arXiv.1205.6904

5.   Crane, T. (1988). The Waterfall illusion. *Analysis, 48*(3), 142-147. https://doi.org/10.2307/3328218

6.   Katz, M., Sohrabi, S., Udrea, O., & Winterer, D. (2018, June). A novel iterative approach to top-k planning. In *Proceedings of the International Conference on Automated Planning and Scheduling* (Vol. 28, pp. 132-140). https://doi.org/10.1609/icaps.v28i1.13893

7.   Matos, P. V., Romão, M., Sarmento, J. M., & Abaladas, A. (2019). The adoption of project management methodologies and tools by NGDOs: A mixed methods perspective. *Journal of Business Research, 101*, 651-659. https://doi.org/10.1016/j.jbusres.2019.01.067

8.   Mills, H. D. (1976). Software development. *IEEE Transactions on Software Engineering, SE-2*(4), 265-273. https://doi.org/10.1109/TSE.1976.233831

9.   Sawyer, S., & Guinan, P. J. (1998). Software development: Processes and performance. *IBM Systems Journal, 37*(4), 552-569. https://doi.org/10.1147/sj.374.0552

10.   Smith, D. (2012). Planning as an iterative process. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 26, No. 1, pp. 2180-2185). https://doi.org/10.1609/aaai.v26i1.8449

11.   Weitzman, M. (1970). Iterative multilevel planning with production targets. *Econometrica, 38*(1), 50-65. https://doi.org/10.2307/1909240

12.   Wells, H. (2012). How effective are project management methodologies? *Project Management Journal, 43*(6), 43-58. https://doi.org/10.1002/pmj.21302

13.   Xing, L., & Chen, G. T. (1996). Iterative methods for inverse treatment planning. *Physics in Medicine & Biology, 41*(10), 2107. https://doi.org/10.1088/0031-9155/41/10/018