PROJECT MANAGEMENT METHODOLOGIES AND FRAMEWORKS USED IN THE SOFTWARE INDUSTRY – CASE STUDY AND COMPARATIVE ANALYSIS

Andreea-Iulia CONCEA-PRISĂCARU[1], Tudor-Alin NIȚESCU[1], Valentin SGÂRCIU[2]

Project management methodologies and frameworks play an important role in the development and successful completion of a project. Choosing a certain framework or methodology is a hard decision and it should be based on a prior and extensive study, taking into consideration several factors (economic, environmental, social and others). In the software development industry, there are many management methodologies available, which are in a constant growth and transformation, adapting themselves to the current needs of the market. This study showcases a thorough research on the software project management methodologies and frameworks available now on the market. The research aims to present and compare different types of methodologies, presenting the advantages and disadvantages of each one, as well as the factors underlying their adoption; being conducted with employees in different roles and from different companies.

Keywords: project management, software development industry, software methodologies, software frameworks, Agile, Waterfall.

1. Introduction

The project management methodologies and frameworks are some of the key factors underlying a successful project. Both concepts refer to a set of rules and guidelines which should be followed in order to standardize, organize and better structure the entire work of a project.

There is a fine line between the two concepts, a methodology can be defined as a set of well-established methods which can be used in a certain field, while a framework can be defined as a set of structured rules or ideas, which can be applied to a certain domain as well. The framework is more flexible, allowing room for creativity and interpretation, while the methodology is more rigid. Moreover, the framework only tells us what to do, while the methodology, besides

[1] PhD student., Dept. of Automation and Industrial Informatics, University POLITEHNICA of Bucharest, Romania, e-mail: andreea.concea@stud.acs.upb.ro.
[1] PhD student., Dept. of Automation and Industrial Informatics, University POLITEHNICA of Bucharest, Romania, e-mail: tudor_alin.nitescu@stud.acs.upb.ro.
[2] Prof., Dept. of Automation and Industrial Informatics, University POLITEHNICA of Bucharest, Romania, e-mail: valentin.sgarciu@upb.ro.

telling us what to do, it is also telling us when and how, providing more details for the entire flow.

Nowadays, the software development field has a difficult choice to make when deciding which methodology or framework to adopt, due to the large number of choices. Choosing a certain methodology or framework should be a decision to be taken after leading an extensive study, which should focus on analyzing the needs of the project and the capabilities of each methodology.

One of the most used methodologies in the software development area is Agile. Agile supports a great range of frameworks like: Scrum, Kanban, XP and others. These are some of the modern and trending approaches, but we shouldn't neglect the traditional ones, from this category we can mention "the old, but gold" Waterfall methodology.

There are many and various factors which can influence the adoption of a certain type of methodology or framework, starting with the company policy and values, budget and costs, time schedule and milestones to be reached, number of employees and others. Therefore, we shouldn't rush when taking such an important decision, and all factors, from major to minor ones, should be considered when reaching a conclusion on this matter. We are going to discuss more in detail about some of these methodologies and frameworks, as well as the factors which can influence their adoption, in the next sections.

This paperwork aims to present an in-depth analysis and comparison between different types of management methodologies used in the software development area, with the purpose of easing the choice of future tech companies which plan to adopt a certain methodology. We will be presenting the advantages and disadvantages of each one, and also the factors which can influence this kind of decision. The study was conducted with employees in different roles and different companies from the software industry.

In the next section we will present some relevant findings from the same area of interest, software management, and we will compare the findings with our own findings, in the following sections.

2. State of the art

Project management, as well as its frameworks and methodologies, represented a highly sought-after topic, that has been in the focus of many researchers lately. This topic has been in high demand due to the large and various list of options available now on the market, this thing increasing the level of difficulty of decision making on this matter.

In this section we are going to present some of the most relevant findings from the project management domain. We are going to focus on two case studies

which present the applicability of software management methodologies and frameworks in software projects.

A case study from the Scientific Bulletin of U.P.B. (University "Politehnica" of Bucharest) highlighted the differences between the traditional and the modern methodologies using the results of a survey with people from different companies. The conclusion of their paperwork was that there is no precise choice when choosing a methodology, the decision being based on several factors. They stated that traditional methodologies, such as Waterfall should be used in linear, well-organized projects, with well-defined requirements, where the customer should be less involved in the project phases, while the modern ones, such as Scrum, should be used for small teams, with changing requirements and a lot of interaction with the customer [1].

A study that compares the Waterfall model and the Agile methodologies, by project characteristics, has concluded that Agile is more suitable than Waterfall for small projects due to its flexibility. Both software development models are facing issues when tackling medium-sized projects. For big and complex projects, consisting in multiple teams working in parallel, Waterfall is more suitable than Agile, because Waterfall pays attention to the software architecture in detail, which is very important for large software projects. This paper compared both software development models showing that Agile is better for smaller teams and projects, changing requirements, interpersonal knowledge, quickly delivering software, while Waterfall is better for larger teams and projects, with less customer interaction, more documentation and heavily focusing on software architecture [2].

Another study that compares the two software methodologies focused on the differences, advantages and disadvantages of both, from practitioners' point of view. They organized interviews with people from various sectors, company sizes and age groups. One of the main differences between the two models was the planning, Waterfall relying on a holistic approach and planning in detail, while Agile is being more flexible, incremental and short term oriented. Waterfall is focusing on the initial plan, while Agile has a light weight working process, relying on the customer interaction, constant feedback loop and adapting the needs frequently. Some of the advantages of Waterfall found in this study were: well established roles and responsibilities, planning and documentation, more predictable, reliable estimation. On the other hand, the Agile advantages were: adaptability to changing requirements, faster flaw detection and remediation of bugs, continuous optimization of the project processes. The results of the interviews shown that no software development model is good for all types of projects, each one being suitable for a certain type of project, depending on the requirements and resources [3].

These results are going to be compared with our own outcomes on the case study and conclusion sections.

In the next section we are going to present the theoretical concepts of project management methodologies and frameworks. We will focus on presenting the main elements and we will also provide examples for each category.

3. Methodology

3.1. Project management introduction

The main goal of project management is to provide value through products that are being delivered to the end consumers, involving several aspects, such as methodologies, techniques, capabilities, or even tools. The project management domain provides us different types of approaches and frameworks that can benefit the project by improving the flows and processes [4].

3.2. Project management methodology vs framework

A project management methodology is based on several elements: best practices, standards, methods, procedures, all that being applied on a certain project, for a certain domain. A project management framework is also based on several elements, such as: rules, ideas, guidelines, which are being applied as well on a certain project, for a specific domain.

Fig. 1. Methodology vs Framework

Both concepts involve rules and guidelines, which being applied to a certain project are providing value by organizing and better structuring the entire work. Although the two concepts are quite similar, the main difference between

them is the flexibility level, as we can see in the comparison from Fig. 1. The framework allows room for new ideas, creativity, is more flexible, while the methodology is more rigid, requiring stricter rules. This difference can be noticed by comparing the following criteria: structure, standards and trending. The frameworks are growing on the popularity level among the users, by providing more flexibility, giving some main guidelines regarding what is needed to be done, while the methodologies are being less popular, providing in depth details and indications about what and how to do the required work.

3.2. Project management methodologies

Now that we have clarified the project management concepts, we are going to present some of the most popular methodologies and frameworks used in the software development domain. Between the most relevant methodologies used in the tech companies we remember Agile and Waterfall.

3.2.1. Agile

Agile is a broad term used for defining a set of software development principles. It is a theoretical framework applied in the software engineering area, that starts with the planning stage and progresses through incremental and iterative interactions throughout the course of a project lifecycle, until the deployment stage, as we can see below (Fig. 2). Agile approach's main objective is to reduce the overhead in the software development processes by enabling the adoption of adjustments that do not compromise the process or require a lot of extra efforts.

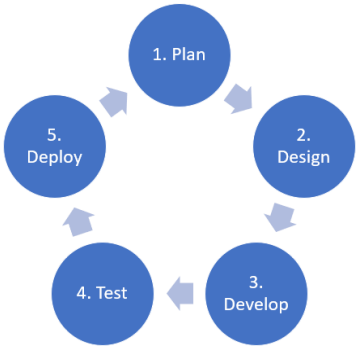


Fig. 2. Agile stages

The term "Agile" is occasionally used inaccurately; it describes a set of practices, values, and guiding principles rather than an approach or methodology. Agility is the capacity to quickly respond to any change in the environment, to the needs of the customers, or to any delivery impediments that might appear. It is associated with ideas of flexibility, rapidity, liveliness, or awareness [5].

The "Agile Manifesto" outlines a set of values and guidelines for software agility, consisting in four values and twelve principles. Following is a list of the

four values supported by Agile: individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation and last but not least, responding to change over following a plan [6].

The Agile paradigm, which has been around for more than 20 years, has had a significant impact on the software development industry due to its lower costs and improved productivity, quality, and satisfaction level. It accomplishes this through its flexible handling methods, as well as enhanced communication and coordination techniques [7].

3.2.2. Waterfall

The Waterfall model is very easy to comprehend, and it can be used to demonstrate the conventional approach for software development. The Waterfall model is a static methodology that takes a linear, sequential approach for the software development process, finishing one task before beginning the next one. Waterfall approach divides projects into stages such as requirement analysis, design, coding, and testing.

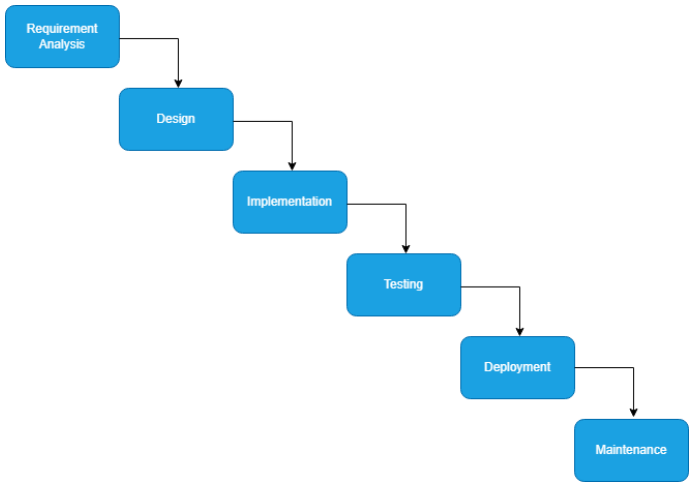


Fig. 3. Waterfall stages

Usually, the activities are the following: communication, planning, modeling, building, and deployment. To summarize this, the stages of the Waterfall methodology are: requirement analysis, design, implementation, testing, deployment and maintenance. Every stage of development has a specific goal in the Waterfall methodology. There is no possibility to go back and reassess an earlier stage, as seen in Fig. 3 from above. Once a phase has been fully completed, we are moving on to the next one [8].

3.3. Project management frameworks

Among the most popular and efficient project management frameworks used nowadays on the software market we count Scrum, Extreme Programming (XP) and Kanban.

3.3.1. Scrum

One of the most used frameworks from the Agile area is Scrum. It allows the programmers to work together in scaled projects, splitting them into smaller parts, parts which can be further accomplished in fixed-duration cycles, or "sprints", while tracking the progress and planning/refining the plans in regular meetings to deliver products incrementally. A Scrum Team has the following components: one Scrum Master, one Product Owner and the Development Team. While using Scrum facilitation, it is typically necessary to enhance teamwork and team motivation, to clarify who is doing what, to support with conflict resolution, and to make sure that all the members are contributing to the team's goal. Just like other members of the team, the facilitator, Scrum Master, works on a task board, which is used to track the project tasks progress. The Scrum Master could be an experienced teammate, or a dedicated professional hired for this purpose. Therefore, the Scrum Master assumes the responsibilities of a team leader or project manager in the traditional sense and is in charge of putting Scrum values and guidelines into practice, as well as removing the eventual impediments [9].

Fig. 4. Scrum team composition

3.3.2. Extreme Programming (XP)

Extreme Programming is a relatively straightforward, adaptive development process that can manage unclear, ambiguous, or rapidly changing requirements, designed to work well for small and medium size teams. Extreme Programming is a set of principles, rules and techniques that are being applied in a systematic way. It is the outcome of pushing the boundaries of techniques that proved to be effective in producing high-quality software. The importance of customer satisfaction in this Agile methodology is crucial. Frequent reports and releases of partially finished software are required to address flaws and problems at an early stage. Extreme Programming involves working in pairs and shifts, for example: one programmer is writing the code and the other one is watching the process and vice versa. This technique proved to be effective, foreseeing the possible issues and flaws that might appear, correcting them in time, because two

pair of eyes are better than one. A smaller amount of flaws reduces the development costs and expenses, producing high-quality products at reduced prices. The phases of development while using XP are the following ones: exploration phase, planning phase, release planning phase, iteration planning phase, iteration phase, release phase, production phase, maintenance phase, death phase [10].

3.3.3. Kanban

Kanban is a term which comes from the Japanese word used for "visual signal". It was implemented initially by the workers of Toyota, and it was designed with the purpose of keeping track of the processes in their production system. Thanks to this technique, the teams were able to communicate more efficiently, foreseeing the things that are needed to be done. As a short definition, Kanban tells you what is required and how to achieve it, all of that in an organized and visual manner. Kanban's ability to clearly visualize each developer's allocated work may be its most valuable feature. It also clearly describes priorities and reduces work in progress by only producing items that have been specifically requested, which promotes continuous supply of items. The key goals of Kanban are the flow of work and the removal of unnecessary operations, both of which result in shorter feedback loops. Eliminating waste, enhancing quality, producing knowledge, delivering software quickly, and optimizing are just a few of the Kanban guidelines. The Kanban board is a visual tool used to organize the work of a Kanban team. It is useful in providing the current status of the tasks, improving the predictability and communication at team level [11].

In the following section we are going to present a case study and a thorough analysis based on the obtained results.

4. Case study and results analysis

We conducted a case study among employees in different roles and from different tech companies in order to observe the practical sides of using a certain methodology. The survey was focused on Agile and Waterfall on the methodology side. With this opportunity we wanted to highlight the advantages and disadvantages of the methodologies stated above, as well as presenting a comparative analysis based on the obtained results. Furthermore, we wanted to present the factors underlying the decision to adopt a certain methodology, as well as showcasing some real-life examples of usage from the participants experience point of view.

The number of participants was 156. We had participants in different roles: Developer, Scrum Master, Product Owner and Project Manager. The participants were from different domains of the tech field.

The participants level of experience was well distributed, from juniors to seniors, the majority having between 1-3 and 3-5 years of experience, as we can see in Fig. 5. The percentages obtained were the following ones: 6.82% of the participants had less than 1 year of experience, 36.43% of the participants had between 1 and 3 years of experience, 44.78 % of the participants had between 3 and 5 years of experience, while 11.97 % had more than 5 years of experience.

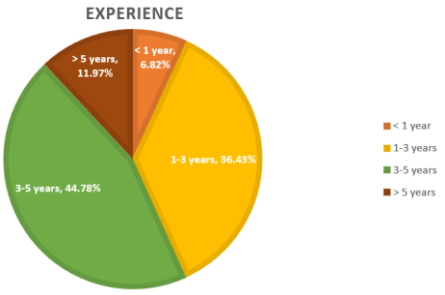


Fig. 5. Participants experience

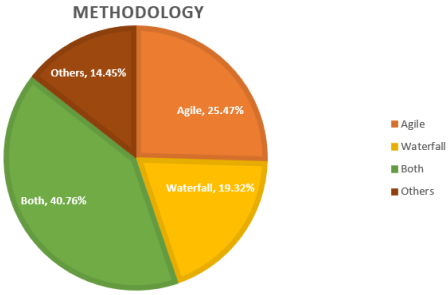


Fig. 6. Used methodologies

The participants have worked with Agile, Waterfall and other methodologies. The results were the following ones: 25.47% for Agile, 19.32% for Waterfall and 40.76% for both Agile and Waterfall, while 14.45% have worked with other methodologies. The results have shown that the majority has worked either with Agile, either with both Agile and Waterfall as methodologies (Fig. 6).

Between the participants that have worked with Agile, the majority has worked with the Scrum framework, Kanban and XP being less popular among them. The results were: 57.88% of the participants have worked with Scrum, 23.21% of the participants have worked with Kanban, 11.52% of the participants have worked with XP, while 7.39% have worked with other frameworks (Fig. 7).

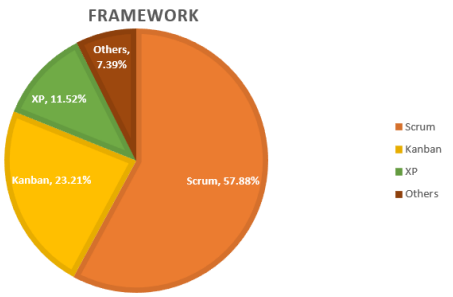


Fig. 7. Used frameworks

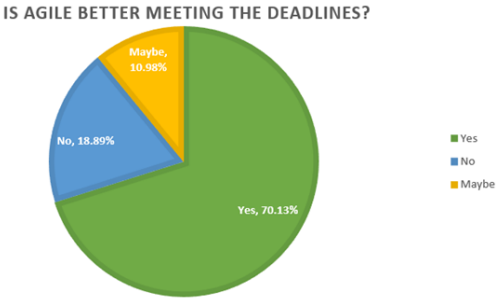


Fig. 8. Agile meeting deadlines

Furthermore, 70.13% of the participants using Agile said that this methodology respects the deadlines, while 18.89 % disagreed and 10.98% were in

between, as we can see above in Fig. 8. On the other side, comparing the two methodologies, Agile with Waterfall, it seems that Agile is more effective on achieving deadlines in time and respecting the milestones, while on the Waterfall side the results were the following ones: 56.42% agreed that Waterfall is respecting the deadlines, 21.39% disagreed and 22.19% were in between. Analyzing these results, the factor of success of Agile on this topic is due to the iterative and incremental approach, being more flexible and measurable.

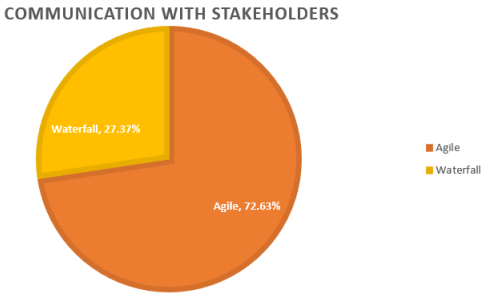


Fig. 9. Communication with stakeholders

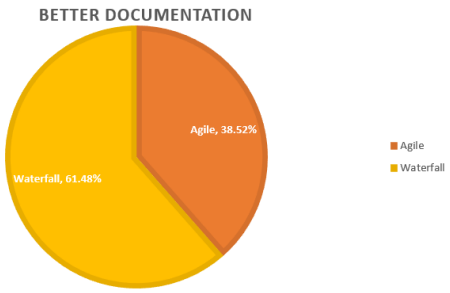


Fig. 10. Better documentation

Due to the same level of flexibility and adaptability to the customer needs, it seems that Agile is more effective from the communication with stakeholders point of view, the outcome of the survey being the following one: 72.63% consider that Agile is more effective on the communication side, while 27.37% consider Waterfall is (Fig. 9).

On the other side, Waterfall seems to gain at the documentation level, 61.48% consider that projects using Waterfall methodologies are better documented, wile Agile obtains only a percentage of 38.52% (Fig. 10). Wanting to compare the two methodologies we created a top 3 advantages and disadvantages for each one, as you can see below.

Top 3 Agile advantages are:

- Collaboration inside team;
- Adaptability to changing requirements;
- Feedback loop.

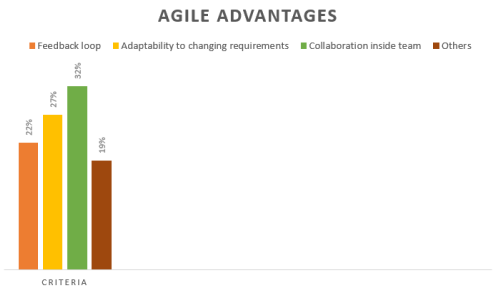
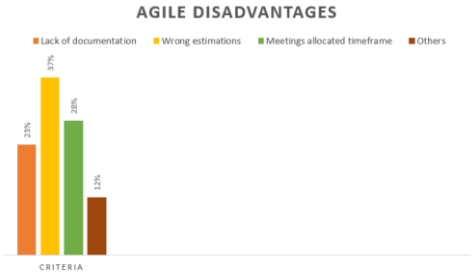


Fig. 11. Agile advantages

Top 3 Agile disadvantages are:

- Wrong estimations;
- Meetings allocated time;
- Lack of documentation.

Fig. 12. Agile disadvantages



Top 3 Waterfall advantages are:

- Documentation;
- Well defined requirements;
- In depth testing.

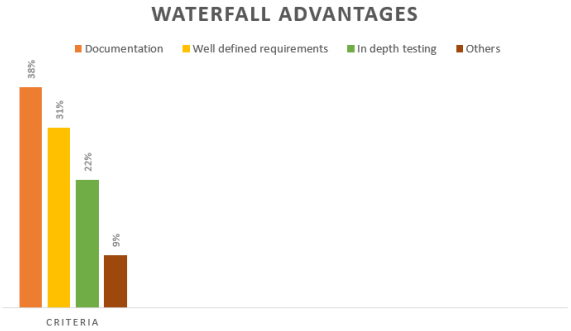


Fig. 13. Waterfall advantages

Top 3 Waterfall disadvantages are:

- Changing requirements;
- Lack of client feedback;
- Meeting deadlines.

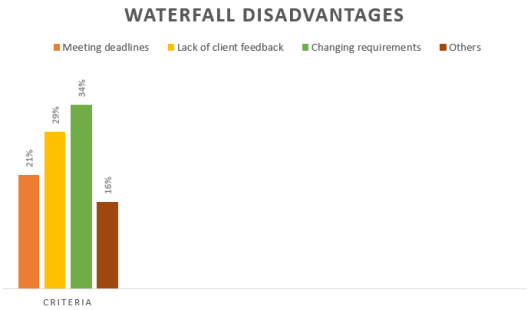


Fig. 14. Waterfall disadvantages

Based on the obtained results, we wanted to highlight, besides the advantages and disadvantages of each methodology, also the factors that underly the decision of adoption of each one, through a comparative analysis.

Therefore, we can say that **time** and **costs** are two important factors that can influence the decision of choosing a methodology. Using Agile, the costs of flaw remediation are lower due to the integration of testing since the earlier stages of the software development lifecycle. When speaking about Waterfall, the costs are higher, the testing taking place only later on, towards the end of the project. When using Agile the flaws can be spotted and fixed since the very beginning, while when using Waterfall, they can be spotted and fixed only at a later phase, this increasing the remediation costs. On the other hand, when taking into consideration the time criteria, Agile seems to be better than Waterfall in **meeting the deadlines**, this being due to its incremental and iterative approach, maintaining a tight relationship with the stakeholders and requiring constant feedback from their side. Waterfall is losing this category due to its lack of adaptability to changing the requirements. Another important factor is the **human resources** one, to be more precise, the team and its composition. Here we can mention that we noticed a better communication level on the Agile side, the level of communication and collaboration inside the team seemed to be higher in this case. This can be due to the smaller number of team members and to the recurrent meetings. On the Waterfall side the number of team members is usually higher, and the meetings are not happening that often, being harder to maintain a tight relationship between the team members. While Agile is winning on several chapters, Waterfall also has some strong points, here we can mention the **documentation** and **in-depth testing phase**. When speaking about documentation, Waterfall projects are very well documented, being easier for a newcomer to understand what is happening and to get used to the project. This thing is being reinforced by the in-depth testing phase, which is happening in the final stages of the project, this process involving also documenting very well the project and everything that was tested.

In the following section we are going to present the conclusions that we reached after analyzing the results that we obtained from the case study.

5. Conclusions

In conclusion, our paperwork presented some of the most used and controversial methodologies from the tech market, Agile and Waterfall. Our case study was focused on presenting the advantages and disadvantages of each methodology, as well as a comparative analysis based on the obtained results, which helped us identify the factors which are underlying their adoption, this part highlighting our personal contribution.

The case study results seem to be in favor of Agile methodology. Some of the factors that led to these results were: the high level of adaptability, constant feedback loop and communication with stakeholders, increased communication and collaboration inside the team and lower flaw remediation costs. When choosing a methodology for a new project, we should take into consideration several factors, such as: time, costs, human resources, documentation and others.

Some common grounds that we found when comparing our findings with the ones obtained in the case studies from section 2, the results are in favor of Agile methodology when speaking about smaller projects, with rapidly changing requirements, while Waterfall is more suitable for long term projects, as it relies more on well-defined requirements and documentation. Besides that, Agile is better for interpersonal knowledge and communication, stimulating motivation and collaboration within the team. Moreover, Agile is more focused on customer interaction and feedback loop, while Waterfall tends to be more rigid.

Therefore, based on the results, we can state that Agile methodology is better suited for a larger variety of projects, due to its flexibility, tendency towards teamwork and constant interaction with the customers. Although Agile seems to have more advantages, we shouldn't neglect Waterfall, as it has its own strong points, especially when talking about documentation, well defined requirements and in depth testing phase. Having this said, the choice between the two of them should be made based on the project requirements and available resources, taking into account the advantages and disadvantages presented above.

Our future direction is to realize a case study which will focus more on the framework side, analyzing the same aspects and concerns, as well as their integration with the methodologies.

R E F E R E N C E S

[1] *I.I. Anghel, R.Ș. Călin, M.L. Nedelea, I.C. Stănică, C. Tudose, C.A. Boiangiu*, "Software development methodologies: a comparative analysis", U.P.B. Scientific Bulletin, Series C, Vol. 84, Iss. 3, 2022.

[2] *W. Van Casteren*, "The Waterfall Model and the Agile Methodologies : A comparison by project characteristics", Research Gate, 2, pp.1-6, 2017.

[3] *T. Thesing, C. Feldmann, M. Burchardt*, "Agile versus waterfall project management: decision model for selecting the appropriate approach to a project", Procedia Computer Science, 181, pp.746-756, 2021.

[4] *P. Jovanovic, I. Beric*, "Analysis of the available project management methodologies", Management: Journal of Sustainable Business and Management Solutions in Emerging Economies 23.3, pp 1-13, 2018.

[5] *S. Al-Saqqa, S. Sawalha, H. Abdel-Nabi*, "Agile Software Development: Methodologies and Trends", International Journal of Interactive Mobile Technologies, Volume 14, Number 11, 2020.

[6] *P. Rodríguez, M. Mantyla, O. Markku, L. E. Lwakatare, P. Seppanen, P. Kuvaja*, "Advances in Using Agile and Lean Processes for Software Development", Advances in Computers, Volume 113, pp. 135-224, Elsevier, 2018.

[7] *A. Rauf, M. Al-Ghafees*, "Gap Analysis between State of Practice and State of Art Practices in Agile Software Development", Agile Conference IEEE, pp. 102-106. 2015.

[8] *A. A. Adetokunbo, B. A. Adenowo*, "Software engineering methodologies: a review of the Waterfall model and object-oriented approach", International Journal of Scientific & Engineering Research, 4(7), pp 427-434, 2013.

[9] *E. S. Hidalgo*, "Adapting the scrum framework for agile project management in science: case study of a distributed research initiative", Heliyon 5.3, Elsevier, 2019.

[10] *A. Akhtar, B. Bakhtawar, S. Akhtar*, "EXTREME PROGRAMMING VS SCRUM: A COMPARISON OF AGILE MODELS.", International Journal of Technology, Innovation and Management (IJTIM) 2(2), 2022.

[11] *W. Zayat, O. Senvar*, "Framework study for agile software development via scrum and Kanban", International Journal of Innovation and Technology Management, 17(04), 2030002, 2020.