International Journal of Pioneering Technology and Engineering

Research Article

A Model Proposal for Scaling the Productivity Increase in Agile Project Management Methodology

Gürkan Işık [a*] **Mert Can Çifci** [a]

[a] Department of Industrial Engineering, Bursa Technical University, Bursa, Türkiye,

Article Info

Article history
Received: 31/07/2023
Revised: 04/09/2023
Accepted: 21/11/2023

Keywords:
Project management,
Agile methodology,
Waterfall model,
Project productivity.

ABSTRACT

Project management in the software industry is the process of successfully planning, managing, and completing software development projects. Software projects can often be complex, constantly changing, and full of uncertainties, so project management is of particular importance in the software industry. There are two basic approaches to the execution of software projects. In the waterfall model, the project stages are operated in the determined order and after each stage is completed, the next stage is passed, while in the agile approach, the project is divided into small and priority parts, and the work processes are repeated at regular intervals and improvements are made depending on the feedback received. Recently, there has been a trend towards managing software projects with agile methodologies, as it provides flexibility and adaptability. However, there is no quantitative evaluation of the benefits of agile structuring both in the literature and in the sector, and this benefit is expressed with verbal expressions referring to the formation of a dynamic culture. For this reason, the productivity increase brought by the agile project approach cannot be measured and the benefit it provides cannot be scaled. In this study, a model that takes linguistic uncertainty into account is proposed to quantitatively scale the productivity increase provided by agile project structuring. The proposed model has been applied in a business where enterprise resource planning solutions have been developed and the effect of the transition from waterfall model to agile methodology on the productivity of the project has been investigated. Within the framework of the procedure proposed by the model, expert evaluations were taken in line with the determined quantitative and qualitative criteria, these criteria were listed with the fuzzy SWARA method, and the productivity increase provided by agile methodology was calculated using this information. The findings also allowed us to get an idea of the point of scaling the increase in productivity across the industry.

1. Introduction

Along with the development process of projects in the software sector, many methodologies have emerged, and these methodologies have been applied in different projects. However, over time, some deficiencies or implementation difficulties were faced in used methodologies, and this necessitated the renewal of methodologies or development of new methodologies. Today, software plays an important role in many fields from homes to defense systems and supports business processes of various industries, thanks to rapidly developing technologies and concepts such as cell phones and computers. Development of a such software is carried out with certain methodologies, considering the factors.

The constantly changing needs, technologies and complexity of projects have allowed methodologies to evolve and have led to the emergence of more adaptive, effective, and flexible methods. In this process, new approaches and frameworks were developed in line with experiences and feedback, and it was aimed to manage projects more successfully. Adebayo & Lassi [1] identified three different profiles of project managers in terms of the quality of teamwork and the type of project management (PM) they use: pure agile, hybrid that leans towards the waterfall model, and hybrid that leans towards agile methods. These approaches have certain advantages compared to each other. For example, if the requirements are well defined, the business functions are well designed, the waterfall model can yield a lean management. On the other hand, if uncertainties are high, the agile methodology can reduce the failure risk of the project.

In the studies discussing advantages of PM methodologies, evaluations are made that the agile organization structure is more efficient in general. The advantages of agile structuring are listed, and it is emphasized that it is a methodology that better meets current needs, especially in the field of software development (SD). However, in all of these studies, benefits of agile methodology are stated with verbal expressions. There is no mathematical comparison between the costs caused by the difficulties of the method and the gains brought by the benefits it provides. For this reason, the productivity increase brought by the agile project approach cannot be measured and the benefit it provides cannot be scaled. One of the most important reasons why such a comparison cannot be made is that the majority of the criteria subject to comparison are qualitative factors. Measuring and comparing qualitative factors is not possible with traditional methods. There is a need for an approach that consider the uncertainty inherent in qualitative evaluations and allow computing with words. In addition, the approach to be used must be in a structure that can take both qualitative and quantitative factors into account.

In this study, to quantitatively scale the productivity increase provided by agile project structuring, a model built on a logic considering negative and positive ideal situations and the linguistic uncertainty caused by the qualitative factors is proposed.

***Corresponding author:** Gürkan Işık
*E-mail address: gurkan.isik@btu.edu.tr
https://doi.org/10.56158/jpte.2023.48.2.02

The model consists of 3 main steps. In the first step, criteria set and linguistic terms that will represent fuzziness are determined. In the second step, criteria are weighted with the fuzzy linguistic SWARA method. In the 3rd step, numerical conversion, scaling, and efficiency calculations are made, the results are compared to ideal solutions, and the productivity increase is obtained in percentage. Finally, the final productivity score for the PM method is determined. The proposed model was applied to a company developing enterprise resource planning (ERP) solutions. The productivity increase obtained in the transition from waterfall model to agile methodology for this company is scaled. It is thought that findings of this study form a reference point for scaling the increase in productivity across the industry.

The rest of the paper is organized as follows: project management approaches in software development are summarized and compared in Section 2, the proposed model is presented in Section 3, and is applied to a firm in the ERP sector in Section 4. Conclusions and future research directions are given in Section 5.

2. Project Management and Software Development Process

Projects are temporary efforts to meet a specific need [2]. The project aims to achieve a specific goal. This goal may be to develop a product, provide a service, improve a process, or achieve a specific result. The project provides the transition of the organization from its current state to another state and is completed within a certain time frame [2]. A project can be defined as a temporary effort and that produces a distinctive product, service, or result. A project is a process carried out using limited funding and resources for a specified period to solve a problem or seize an opportunity. While considering customer satisfaction and quality, possible risks are managed, and a unique plan is followed to achieve the determined goals and objectives. Every project has a beginning and an end. Projects are tools used by organizations to achieve their strategic goals. Projects are planned and implemented to move organizations from one current state to another [2]. Project objectives are important steps that transform problems into objectives and direct them to activities and are shaped within a framework that includes elements such as environmental factors, target audience, public authorities, financing institutions and environmental conditions [2] . Projects should differ from routine work and aim for change. Each project is unique and has a specific start and end time. The place where the project will be carried out should be specific and defined. Projects are carried out with limited resources and are planned. The project needs certain resources and a budget. The evaluation criteria of the project are determined at the beginning of the project and the evaluability of the project is ensured. The project result must be requested by a prominent user or client. People and the workforce are important components of projects and play an influential role in the success of projects [2] .

2.1. Project management

PM is the management of activities organized to achieve a common goal, in which special management structures and techniques are used to achieve a specific goal. In the past, it emerged with the increasing complexity of corporate life and was shaped by developing management ideas. Especially the periods when large-scale government projects were carried out contributed to the development of PM. PM is a process that involves planning, organizing, providing, and managing resources for the successful completion of a project. This discipline has emerged from a combination of methods used in construction, engineering, defense, and other industries. The need for PM arises from reasons such as the need for fast completion, dynamic work environments, complex workforce and organizational structures, limited resources, adaptability to innovations, complex communication, and increased information resources. The Project Management Institute (PMI) was established to set standards and promote good practices in the field of PM. PMI published the PM Information Manual (PMBOK) in 1969 and today has grown into an institute with 475,000 members in 207 countries around the world. PMBOK is defined as PM's application of processes, methods, knowledge, skills and experience to achieve objectives. PMI has made adjustments to version 6 (2017) of PMBOK to keep up with changes in the industry [3].

In order to manage projects successfully, resources in the project should be used efficiently, risks should be minimized, communication and cooperation should be strengthened, and customer satisfaction should be increased [2]. When these are provided, PM provides several advantages. More economical development processes are achieved, and work is progressed more efficiently. It enables resources to be used and controlled more effectively, which contributes to lower costs and higher profits. It ensures high quality and safety standards, so customers are offered reliable products and services. It provides effective coordination and motivation and increases cooperation between team members. It improves customer relations and increases customer satisfaction. It predetermines the duration and methods of achieving the goals and objectives of the project. It minimizes the need for continuous reporting and facilitates communication and information flow. It helps to predetermine the time required for the project and provides time management. It anticipates project-related costs and facilitates budget planning. It explains the determination of the necessary resources and the technology to be used. It helps to establish control systems and ensures the regular monitoring and management of the project. It ensures that all tasks are shown in organizational charts, and roles and responsibilities are clearly defined. Helps project team members develop PM skills, and enables projects to be carried out more successfully [2]. On the other hand, there are some risks brought by PM. For example, those who manage projects must be successful in human relations otherwise PM can lead to the emergence of complex organizational structures, tending to the deterioration of organizational policies [4]. An ill-structured project can lead to management difficulties; it is necessary to manage the project effectively. Staffing can be difficult, appropriate skills and management of resources are essential [4].

Projects often go through processes that involve similar stages from start to finish. This management function covers the phases of project initiation, planning, implementation, follow-up, control, and closure. These processes are called the main processes of the project [5]. The inception phase is the first phase in a project's lifecycle. At this stage, the project is defined, funding sources are determined, the realization of the project is decided, the project team is formed, and job descriptions are determined. Together with the stakeholders, the objectives, constraints, and risks of the project are determined. The feasibility of the project can be evaluated by making feasibility studies specific to the project. Especially in information technology projects, requirements are usually collected and analyzed at the initial stage. The planning phase is the phase where the project is planned in detail. Determining the scope of the project, clarifying the objectives, and defining the sequence of actions required for the project to achieve its intended objectives are important steps in the planning process. In the project life cycle, the planning stage is the most critical, because at this stage the sequence of activities, their responsibilities and their duration are determined. The project manager is tasked with involving the right people in the project and accurately setting the project activities and time frame [6]. The execution phase is the third phase of the project life cycle and aims to achieve outputs in line with the time and resources of the project to perform the planned activities. At this stage, the implementation of the project begins, and the planned activities are completed by the relevant persons or units within the determined time and resource limits. The project team manages project activities, allocates resources, communicates, and monitors project progress. These processes generally include coordinating people and resources, managing stakeholder expectations, and integrating the activities outlined in the project management plan. A large part of the project budget is spent on the execution process. At this stage, the tasks required to achieve the objectives are fulfilled and the progress of the project is continuously monitored, thus ensuring the successful completion of the project [6]. The monitoring and control phase are important tools used to regularly monitor the progress of the project, analyze its performance, and make necessary adjustments. These processes are carried out in order to ensure that the project is progressing by the planned targets, to detect possible deviations and to take corrective or preventive measures when necessary. The project manager regularly collects data based on the determined indicators and evaluates this data on the compatibility of the project with the plans. If deviations from the plans occur, the project manager takes the necessary decisions and even renews the plan of the project. Control and monitoring at this stage are one of the most challenging tasks of the project manager, as continuous monitoring and analysis are required to manage changes and bring the project to plan. These processes are vital to ensure the success of the project and achieve the objectives [6]. The closing phase includes the important steps required to formally complete a project. This process covers the activities necessary for the successful conclusion of the project and the acceptance of its results. Closing steps should be carefully planned and implemented in order to increase the success of the project and customer satisfaction. These steps include assessing the status of the project, properly closing existing resources, and learning the lessons of the project [6].

2.2. Software development methodologies

SD lifecycle (SDLC) is a framework that covers the process from the beginning to the end of software projects. This cycle includes the planning, design, development, testing, deployment, and maintenance of software. Each phase focuses on specific goals, processes and outputs and is managed to ensure the successful completion of the project. SD methodologies are systematic approaches used to plan, implement, and manage software projects. These methodologies include elements such as streamlining the software process, team collaboration, efficient use of resources, and quality assurance. Different methodologies focus on different principles, processes, and tools. Traditional plan-based methodologies build the project process on step-by-step planning and progress control. On the other hand, agile methodologies focus on responsiveness, flexibility, and customer collaboration. Each methodology has advantages and disadvantages, and the choice is made depending on the project's needs and characteristics. SD methodologies play an important role in the realization of successful and productive projects. In this section, information about the waterfall model and agile methodologies will be presented [2].

2.2.1. Waterfall model

The waterfall model is the oldest and most basic model among the known SD methodologies [2]. This methodology is an approach where a project is divided into different phases and used as a sequential PM methodology. In this methodology, each stage starts only after the previous one has been completed. The waterfall model divides the project into distinct and sequential phases, and each new phase begins with the completion of the previous phase. This methodology is one of the traditional methodologies and relies on stringent processes used in industries such as construction and manufacturing. The focus is on an approach to creating the best final product where changes are difficult to make after the project is complete. The steps applied in the waterfall model are shown in Figure 1.

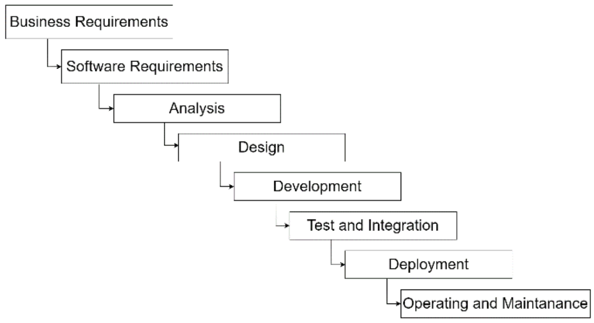

Fig. 1. Steps of the waterfall model

First of all, the outputs obtained after the completion of the requirements analysis stage, or the information obtained at this stage are transferred to the software design stage. In the software design phase, before the coding phase, the design is made, and the output of the design is transmitted to the coding phase. In the coding phase, coding is done according to the selected SD language. Then, the test environment is passed for product testing activities. Various tests are performed in the test environment and the findings are transmitted back to the coding stage. If any problems are detected during the coding or previous design phases, the finding is sent back to the software design phase, necessary corrections are made, re-coding and testing phases are started. Necessary infrastructure preparations are made to put the software into practice, which successfully passes all tests, and training is provided to the users. Then, the software operation and maintenance phase are started. At this stage, new functions can be added during the operational phase. In case of any error, it is returned to the beginning during the operation phase, the cause of the error is found, and the relevant changes are made. After the error is corrected, the software is implemented and put into operation without any errors [4]. While this process is systematically correct, a result that is quite different from the desired result may result. The biggest factor in this is time. The length of the process may change the preferences or expectations of the product owners and cause the product to lose its functionality and market power with industry and technology variability. On the other hand, misinformed information or a different perception by the software developer may lead to the emergence of a completely wrong product. With changing requirements as the project completes certain phases, the project may have to be reconsidered. The waterfall model provided relief from this burden without changing the requirements. However, in this case, a product could emerge that moved away from the change trends and moved away from the vision of the product owner. Given the output-input relationship and the high cost to the project of potential changes, the waterfall model does not allow for phase jumps or reversals. In projects, each phase is marked with completion points, and changes are managed through stringent controls, detailed documentation, and management approval. Implementing this model requires discipline. The waterfall model adopts a design philosophy where thorough analysis and design are prioritized and completed before coding. System and software requirements are defined at a stage where everything is known from the outset and is assumed to remain unchanged. Therefore, it is suitable for projects that do not contain uncertainty and are less likely to change. While the customer is involved in the project in the process of determining the requirements, it stays in the background in the analysis, design and coding processes and reappears in the testing and integration stages. In the waterfall model, issues such as planning, timelines, target dates, budget, and rollout of the system are emphasized. Considering the time and conditions in which this methodology stood out, business processes were not dependent on information systems like today and projects did not need to be terminated quickly. Technological progress and change in business life were not as rapid as today, so projects were not progressing at the same pace. Information systems were hardware-oriented due to technology and resource constraints, and the main purpose of the software was to make the most efficient use of limited hardware resources [2]. Therefore, it can still be considered useful for business areas where similar conditions apply. Clear, understandable, and complete determination of requirements at the beginning of the project helps to make accurate time and cost estimations and reduces uncertainties in the project. It is well known in theory as it is a frequently used methodology. Since it is a simple process, it can be applied easily. It is ideal for inexperienced project members, project managers and teams with fluctuating performance, as the roles and responsibilities of the participants are clearly defined. When an accurate estimate of time, cost, and effort is made, the risk of exceeding resources is minimal. It's easy to keep track of the progress of the project as one phase is completed and the next one moves on. At the same time, the absence of simultaneous phases facilitates resource tracking. Continuous documentation prevents the retrospective documentation process of the project, allowing full documentation of the current situation and facilitating the evaluation of the project [2]. On the other hand, if there is flexibility and resistance to changes, changes can be expensive in terms of time and cost, the client is less involved in the design and development processes, there may be a lack of forward visibility because there may be uncertainties in factors such as project progress and deadlines, early detection and management of risks it can be difficult, returns and changes can be limited [2].

2.2.2. Agile methodology

Agile SD approaches represent the application of lean principles, which emerged with the aim of increasing productivity in the manufacturing sector, the information technology, and the software sector. Agile approaches have started to be seen as various methodologies in the software industry since the 1970s [7]. In recent years, it has been observed that agile approaches have gained popularity around the world and have been successfully applied in many software companies and projects. The Agile SD manifesto is an SD methodology that emphasizes the need for continuous customer collaboration and rapid response to change, with a focus on project teams and interaction between them. This methodology is based on the idea that the software must always be up and running and in close collaboration with the customer [8]. Agile methodology is a modern and simple approach that has been developed as a solution to the disadvantages of previous methodologies and can be applied in various fields and in different ways. True to its name, Agile emphasizes the early delivery of the product and allows for flexible and adaptable changes at any stage of the project. It adopts a collaborative approach to dividing large projects into smaller parts to make them more manageable. Realizing that pre-planned tools are limiting and slow, leaders in information technology and SD have sought alternative approaches to these tools that do not provide the ability to adapt quickly to changing projects. Alternative approaches that emerged as a result of these searches have turned into agile thoughts over time. SD activities in agile methodologies progress through iterative stages. As a result of these iterations, a usable product is created and the development processes are re-done in case the requirements change, taking into account the feedback from the customer or users. These iterations are continued to produce exactly the product the customer wants. The stages applied in agile methodologies are shaped depending on the characteristics and requirements of the project. These stages are shown in Figure 2 [2].

Agile methodologies are divided into various sub-branches with different processes. Different types of agile processes have emerged to realize the agile process over time, and these can be considered as a meta model. There are widely used agile methodologies such as Extreme Programming (XP), Scrum, Agile Unified Process, Feature Driven Development (FDD), LEAN Development, Dynamic System Development Methodology (DSDM), and Microsoft Solution Framework (MSF). XP and Scrum are the most widely applied of these methodologies.

The application rates of agile methodologies are given in Table 1 [9]. The most popular methodology both in the world and in Türkiye is Scrum. Besides, XP and Kanban stand out compared to other methodologies. The Scrum methodology is generally used with Kanban and XP, and organizations prefer hybrid methodologies tailored to their needs. In this way, the use of agile methodologies varies. The scrum methodology, which is the type of agile methodology used in this study, is discussed in the rest of the paper.

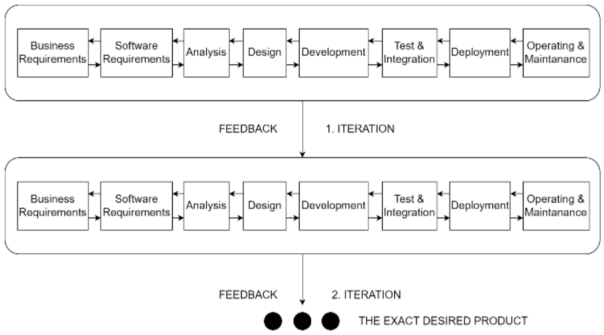

Fig. 2. Steps of agile methodology

Table 1. Application rates of agile methodologies [9]

Agile Methodology	Ratio
Scrum	65%
XP	7%
Kanban	32%
Scrum/XP Hybrid	8%
Scrum/Kanban Mix	7%
Multiple Mixed Adaptations	8%

2.2.3. Scrum

Scrum is a project management methodology used for the agile SD process. It was proposed [10] in the mid-1990s. Its name comes from a term in the game of Rugby and is short meeting where players get together to exchange ideas about the next move. Scrum is a process used to manage and control complex projects. Scrum takes the iterative approach. The project is divided into repetitive processes (sprints) at certain time intervals. Each sprint includes a series of planning, development, testing and evaluation activities [11]. Sprints usually last between 2 and 4 weeks. Scrum is an approach where teams organize themselves and encourage communication and collaboration. The team sets goals at the start of the sprint, schedules the work themselves, and delivers a usable version of the product at the end of the sprint. Communication and progress are tracked through daily scrum meetings [11, 12]. The scrum life cycle is shown in Figure 3 [12].

There are three basic roles in the Scrum methodology: Product owner, scrum master, and development team. These definitions are grouped under the title of roles. The product owner is the business owner or representative of the project. Responsible for understanding customers' needs and determining requirements. The Product Owner determines the priorities of the product, plans the future direction of the product, and monitors the progress of the product's development process. S/he also clarifies requirements, answers question, and provides feedback, in constant communication with the development team [8]. The Scrum Master (SM) is a guide that ensures the smooth functioning of the Scrum process. Teaches team members to scrum principles and practices, supports the team in continuous improvement, and removes obstacles. The SM manages planning meetings, ensures sprint goals are met, and facilitates communication between the team and stakeholders. It also tracks metrics and progress tracked by the team [8]. The Development Team (DT) consists of people involved in the design, development and testing of the product. The team works together to achieve the sprint goals. Each DT member uses their abilities to get the job done. The team is self-directed, plans work, monitors progress, and takes responsibility. DT meets regularly during the sprint to evaluate progress and present work completed at the end of the sprint[8].

Scrum's outputs include the Product Backlog document, the Sprint Backlog document, the Burn down Chart, and the Working Product list. These deliverables are used to manage the requirements and progress of the project, share it with the team, and provide ongoing value [8]. The Product Backlog document is in the form of a list containing all the requirements of the project. This list, created by the product owner, refers to the features and functions of the system in order of priority. It is updated during the project, new features are added, and priorities can be changed[8]. Projects managed by the Scrum methodology are divided into short-term periods called sprints. At the start of each sprint, the project team and the Scrum Master identify the features and functions to be developed in that sprint. Selected features and functions are taken from the Product Backlog and a list called Sprint Backlog is created. The project team only focuses on this list during a sprint[8]. The Burn down Chart shows the status of targeted and realized jobs over time. On the vertical axis, the score totals of the jobs are located, and on the horizontal axis, time. Burn Down and Burn Up charts summarize project progress and the speed at which jobs are completed. The effects of the changes can be seen in these graphs [8]. It is a tested, working product that emerged as a result of the Scrum process. Scrum aims to deliver a functional and valuable product at the end of each sprint by constantly improving the parts of a working product. In this way, the product can be continuously improved based on customer feedback.

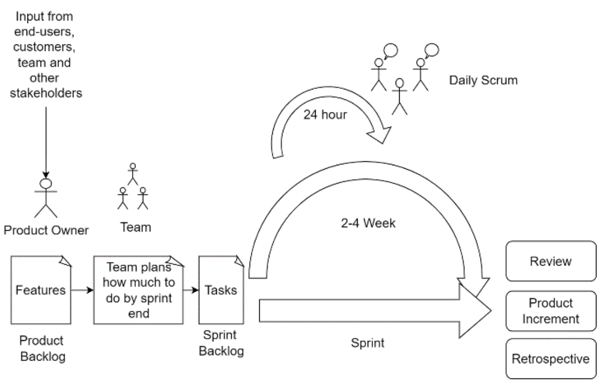

Fig. 3. Scrum lifecycle [12]

In the Scrum methodology, meetings are important communication activities where project teams meet regularly to manage the project process and work in collaboration. In Scrum, different meetings are held for different purposes. The sprint planning meeting is part of Scrum and is a meeting held at the beginning of each sprint. In this meeting, team members and product owners come together to determine the work to be done during the sprint. The product owner shares the sprint goal and priorities, while the team selects jobs and determines estimated completion times. At the end of the meeting, a sprint plan is created and the work to be carried out during the sprint is planned. The sprint planning meeting is important as the team organizes work, clarifies goals, and supports sprint success[8]. Daily Scrum meetings are short-term meetings held at a predetermined location every day at a predetermined time. In these meetings, team members share what they will do in the next 24 hours. Each member reports what he did the previous day and any obstacles he encountered. The Scrum Master looks for solutions to remove these obstacles [11]. Sprint review meetings are a meeting that takes place at the end of each sprint. The work done during the sprint is reviewed and the service or product produced is evaluated. The purpose of this meeting is to see if the software is progressing in line with the product owner's needs. Sprint retrospective meetings evaluate whether the work completed during the sprint has reached the expected quality and is done correctly. This assessment is important for reviewing the team's performance and for improvement in future sprints [11].

According to the basic principles of the agile approach, people and their interaction are more important than the tools and processes used, working software is more valuable than detailed documentation, collaboration with the customer takes precedence over contract terms, and adapting to changes quickly and flexibly is more important than sticking to a solid plan. it is important [8]. The basic principles that result from this perspective can be listed as follows [2, 8]: Early and continuous delivery of valuable software is the most important priority to ensure customer satisfaction, agile processes are adopted to respond flexibly to changing requirements, working software should be regularly presented to the customer in short time intervals, Owners and software developers should work together every day to ensure communication and cooperation since a team of motivated individuals will be the basis for the success of the project, employees should be provided with the support and trust they need, face-to-face communication should be preferred for effective information exchange within the development team, working software is the key to progress. measure, agile processes support sustainable development, it is important to maintain a steady pace among all stakeholders involved in the project process, technical excellence and good design increase agility, simplicity minimizes unnecessary work and increases productivity, best architectures and designs, self-organizing are the results of the teams, the team regularly evaluates itself, constantly improves and adjusts its behavior for effectiveness and productivity. Agile methodologies provide many benefits by offering flexibility, adaptability, and a collaborative approach to business processes. These methodologies provide benefits in areas such as flexibility and adaptability, feedback and control, prioritization and iterations, code quality and pair programming, project planning and sense of responsibility, a healthy work environment, fast response, and communication [4]. However, some limitations and difficulties may arise during the application of these methodologies. For these reasons, the management of projects should take the constraints of agile methodology into account. Otherwise, remote communication difficulties, compatibility issues with subcontractors, decreased productivity in large and complex projects, difficulty in division, lack of customer or user involvement, difficulties in planning and scope, complexity of documentation, team skills and organization, priority changes, reusability issue, testing and quality problems such as control difficulties and user satisfaction may occur [8].

2.2.4. Comparison of agile methodology and waterfall model

The Waterfall SD strategy is a methodology where all the requirements are gathered at the beginning and the processes proceed in order. However, it has strict rules and difficulties in adapting to changes. It is assumed that the requirements are conveyed perfectly and that there will be no changes. Using it in long-running projects may cause loss of time and it may be difficult to manage risks. Conflicts and communication problems can arise between team members. However, it can provide fast outputs on short-term projects with defined and easy requirements [2].

Agile SD methodology aims to adapt to changing requirements. It allows each company to customize their processes and iteratively take risks into account. Thanks to iterations, presentations are made throughout the process and teamwork gains importance. Communication becomes stronger while documentation becomes less important. The customer must also play an active role and be involved in the processes [2].

Successful completion of a project and achieving the targeted results vary according to the methodology used. Different methodologies have different focus and approach to factors such as PM processes, team organization, communication methods and change management. The success rates obtained according to the methodology applied in software projects between 2011 and 2015 given in the 2015 CHAOS report published by Standish Group International are shown in Table 2 [2]. When the table is examined, the data obtained show that the success rates of projects managed with agile methodologies are approximately 3.5 times compared to the success rates of projects managed with the waterfall model. It is stated that the failure rate is 1 in 3 in agile methodologies. However, this may not always be the case, and agile methodologies are not always more likely to succeed than the waterfall model. Each project has different requirements and needs, so the right methodology must be chosen for project success. Both methodologies represent different approaches with advantages and disadvantages. Selecting the appropriate methodology for the project requires considering factors such as the project's requirements, scale, risks, and team structure [2, 4].

Table 2. Success rates of software projects according to the applied methodology [13]

	Waterfall	Agile
Successful Projects	%11	39%
Problematic Projects	60%	52%
Failed Projects	29%	9%

In the literature, there are various studies that examine and compare the Waterfall and Agile methodologies under some headings that are considered to have an impact on success in terms of PM: process flow structure, costs , outsourcing, quality of software, quality assurance and fault tolerance, value generation, user experience, risk management, project complexity, project size, documentation and modeling, reusability, requirements, deliveries, scheduling, customers, roles, team size, industry knowledge, collaboration, complexity in the project [2, 4, 14] .

According to the comparison performed by Zavlova et al. [15], agile and traditional PM approaches require different human resource management architectures. Ciric et al. [7] conducted a study aiming to guide managers in order to understand the differences between agile and traditional approaches and to integrate the agile method into projects with appropriate strategies. Bojan [16] discussed the differences and comparisons between traditional and agile PM in the service sector. Traditional PM and agile PM talked about different approaches in the service sector. Kaim et al. [17] examined the benefits of agile PM in an environment of increasing complexity and made a cost analysis. Knut [18] compared traditional and agile approaches and evaluated applications in different sectors. Rabia et al. [19] reviewed the use of agile methodology in Information Technology (IT) projects and analyzed the advantages and application areas of the agile approach. Daniel & Amy [20] introduced an agile framework for teaching in Scrum and IT PM classes. Abdallah et al. [21] presented a conceptual hybrid PM model for construction projects. Sanchez et al. [22] examined how the transition from traditional PM to agile project could be improved and proposed a PM maturity model. Pace [23] examined the relationship between the management methodology of projects and project success. The article aims to determine whether there is a correlation between different management methodologies used in various projects and project success. By analyzing the research data, the relationship between the management methodologies and the success criteria of the projects was evaluated. The results of the article attempt to show the impact of a particular management methodology on the success of projects. Dursun & Goker [24] aimed to examine the success factors of PM methodologies and to understand the dependencies between these factors. The study analyzes the relationships between success factors of the waterfall, agile and lean six sigma methodologies with the Fuzzy Cognitive Map (FCM) method. Eren [25] presented a comparison that will guide the selection of the appropriate methodology for projects in today's world where agile methodologies are gaining importance as well as traditional and plan-based methodologies.

A summary comparison based on the criteria mentioned above for waterfall and agile methodologies is given in Table 3. The + sign in the table indicates that the methodology is more advantageous in terms of that criterion. When the table is examined, it is seen that the waterfall model has more advantages in 7 items and the agile methodology in 9 items. In the 6th item, both methodologies are advantageous.

Table 3. Summary comparison of waterfall and agile methodologies in terms of determined criteria

Factors	Waterfall	Agile
Costs	-	+
Outsourcing	+	-
Quality of the Software	+	+
Quality Assurance and Fault Tolerance	+	-
Value Generation	-	+
User experience	+	+
Risk management	-	+
Visibility	+	+
Project Complexity	+	+
Project Size	-	+
Documentation and Modeling	+	-
Reusability	+	-
Requirements	-	+
deliveries	-	+
Time Planning	-	+
Management	+	+
Organizational Structure	+	+
Roles	-	+
Experience and Talent	+	-
Team Size	+	-
Area information	+	-
Complexity in terms of Project Elements	-	+

3. A Model Proposal for Scaling the Productivity Increase in Project Management

In this section, a model for scaling the productivity increase provided by agile organizational structuring and a procedure for operating this model are presented. Some of the criteria discussed are quantitative and some are qualitative. Especially when evaluating qualitative criteria, there are uncertainties caused by environmental effects and human factors. Linguistic fuzzy modeling (LFM) is one of the most efficient approaches to use qualitative information in a numerical decision process. For this reason, linguistic fuzzy expert evaluations were used in the procedure so that the productivity calculation could be done properly. There are several decision making methods used for criteria weighting in the literature integrated with LFM. However, most of these methods require evaluations for each combination of criteria, and they rank criteria in an indirect way. On the other hand, SWARA method can make calculations using only the number of evaluations as many as the number of criteria. Additionally, it does not rank the criteria in an indirect way. Thus, the experts making the evaluation can reflect their opinions in the calculation much more easily. In this study, fuzzy SWARA method was used to weight the criteria due to the its simplicity and successful applications in the literature. The following sub-section contains information about this method.

3.1.Fuzzy SWARA Method

SWARA (Step - Wise Weight Assessment Ratio Analysis) method aims to make the best choice by determining the weights of different criteria and alternatives in the decision-making process [26]. In pairwise comparisons, significance values are usually assigned as multiples of 5. For example, if one criterion is 5 times more important than the other, its value maybe 5. These values are used to determine the importance relationship between criteria [26].

The fuzzy SWARA method combines fuzzy set (FS) theory and weighting techniques. In this method, evaluations are carried out using fuzzy linguistic expressions, not numerical expressions. Each linguistic expression is represented by FSs. FSs are used to express uncertainty and uncertain information [27]. The fuzzy SWARA method is a method that can handle uncertainties in multi-criteria decision-making problems and enrich the decision-making process by using FSs. This method helps decision-makers to obtain more comprehensive results by considering their preferences and uncertainties in complex decision-making processes [27].

The fuzzy SWARA method generally includes the following steps:

Step 1: First, the criteria for the problem and the decision makers are determined for the decision committee. It is assumed that the problem has n criteria (j=1, 2,...,n) and k decision makers (k=1,2,...,K) [28].

Step 2: In this stage, each decision maker evaluates each criteria c_j based on their own knowledge and experience and creates a common ranking. In this ranking, $c_1$it represents the best criterion and the worst criterion c_n[28].

Step 3: Based on the integrated ranking, each decision maker determines the comparative weights of the criteria starting from the second row. Decision makers accept the most important criterion as 1.00 points and determine the points according to other criteria. Points are assigned between 0 and 1 in multiples of 5. The average of the comparative weights for each criterion is calculated according to the decision makers (s_j) [28].

Step 4: For each criterion, a coefficient given by Equation (1) (k_j) is calculated. The coefficient of the most important criterion in the joint ranking $k_j$is assigned as 1[28].

$$k_j = \begin{cases} 1, & if\ j = 1 \\ s_j + 1, & if\ j > 1 \end{cases} \quad (1)$$

Step 5: The weight (q_j) for each criterion is calculated using Equation (2). The weight of the most important criterion in the joint ranking is accepted as 1[28].

$$q_j = \begin{cases} 1, & if\ j = 1 \\ \frac{q_{j-1}}{k_j}, & if\ j > 1 \end{cases} \quad (2)$$

Step 6 : Finally, the criteria weights q_j, calculated in the previous step are divided by the total weight to obtain the final weight (w_j) of each criterion as in Equation (3)[28].

$$w_j = \frac{q_{j-1}}{\sum_{j\in J} q_j} \quad (3)$$

Example 1: Let a company wants to make an investment to a new region and experts are requested to make assessments about the following criteria to decide the importance weights of them: E1-Operating Cost, E2-Maintenance Cost, E3-Installation Cost, S1-Cultural Contribution, S2-Job Creation, S3-Social Acceptance, S4-Brand Perception, T1-Lifetime, T2-Technical Maturity, T3-Process Efficiency. Assume that one of the experts made assessment shown in Table 4 by sorting the criteria according to their importance and assigning comparative weights to them. Final weights are calculated according to the calculation steps given above.

Table 4. Example expert evaluations and SWARA calculations

Criteria Id	Criteria	Importance Order	Comparative Weight (s_j)	Coefficient (k_j)	Weight (q_j)	Final Weight (w_j)
T3	Process Efficiency	1		1	1	22,979%
E1	Operating Cost	2	0,1	1,1	0,9090909	20,890%
T1	Lifetime	3	0,75	1,75	0,5194805	11,937%
E3	Installation Cost	4	0,25	1,25	0,4155844	9,550%
E2	Maintenance Cost	5	0,1	1,1	0,377804	8,681%
T2	Technical Maturity	6	0,25	1,25	0,3022432	6,945%
S3	Social Acceptance	7	0,1	1,1	0,2747666	6,314%
S2	Job Creation	8	0,25	1,25	0,2198132	5,051%
S1	Cultural Contribution	9	0,1	1,1	0,1998302	4,592%
S4	Brand Perception	10	0,5	1,5	0,1332201	3,061%

Since the importance comparison of the criteria in the fuzzy SWARA method is made using linguistic expressions, the results obtained are highly dependent on the size of the linguistic term set (LTS). 2-tuple Linguistic Fuzzy Modeling (LFM) approach was adopted in this study in order to eliminate this dependency and allow interim evaluations. In the 2-tuple LFM approach, a 2-tuple linguistic expression is represented by a linguistic term and an accompanying numeric value called "difference of information" (DOI), a type of shifting modifier. The DOI specifies the measure of the sliding motion of the FSs corresponding to the linguistic term between the antecedent and successor FSs [29]. Calculations for 2-tuple LFM are performed according to the following definitions.

Definition 1: Let $S = \{s_i\} = \{s_0, \dots, s_g\}$ be an LTS,$\propto \in [-0{,}5;\ 0{,}5)$ be DOI, $A_{s_i \in S} = (s_i; \propto)$ be a 2-tuple linguistic statement, $f_{s_i \in S}$ is a corresponding FS for s_i linguistic term $F^-(s_i) = \left\{f_{i_j}^-\right\} = \{f_{i_0}^-, \dots, f_{i_9}^-\}$, $[s_{i-1}, s_i)$ be a set with 10 evenly spaced FSs in the range $F^+(s_i) = \left\{f_{i_j}^+\right\} = \{f_{i_0}^+, \dots, f_{i_9}^+\}$, and $(s_i, s_{i+1}]$a set containing 10 consecutive FSs evenly spaced in the range. The FS corresponding to $A_{s_i\in S}$ is found as in Equation (4) [29]:

$$f_{s_i\in S} = \begin{cases} f^{-}_{10\times(1-\propto)} & ,\propto< 0 \\ f_{s_i} & ,\propto= 0 \\ f^{+}_{10\times\propto} & ,\propto> 0 \end{cases} \quad (4)$$

Definition 2: Let $S = \{s_i\} = \{s_0, \dots, s_g\}$be an LTS, and $\beta \in [0, g]$ be a real number. The 2-tuple linguistic expression $A_{s\in S} = (s_i;\ \propto\in [-0,5;\ 0,5))$ corresponding to the number β is found as in Equation (5) [29]:

$$\Delta(\beta) = \left(s_{i=round(\beta)}; \propto= \beta - i\right) = A_{s\in S} \quad (5)$$

Definition 3: Let $S = \{s_i\} = \{s_0, \dots, s_g\}$ be an LTS, and $A_{s\in S} = (s_i;\ \propto)$ be a 2-tuple linguistic statement. The function Δ^{-1} that converts 2-tuple expressions to their equivalent numerical equivalents $\beta \in [0, g] \subset \mathcal{R}$ is defined as in Equation (6)[29]:

$$\Delta^{-1}(A_{s\in S}) = \Delta^{-1}(s_i; \propto) = i + \propto= \beta \quad (6)$$

3.2.Proposed Model

It is very difficult to calculate the productivity of software projects because the criteria for productivity can be in qualitative and quantitative structures. For this reason, in this study, a computational model has been proposed to evaluate qualitative and quantitative data more healthily. The proposed computational model is operated within the framework of the procedure outlined in Figure 4. In the procedure, the linguistic fuzzy SWARA method is used to weight the criteria within itself based on fuzzy expert evaluations for scaling to be done healthily.

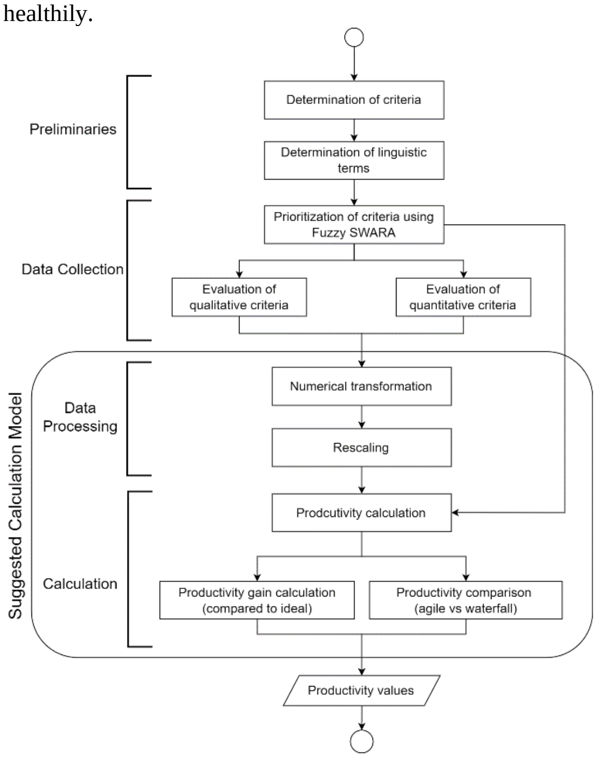

Fig. 4. Suggested productivity calculation model and procedure

4. Application

The proposed procedure was implemented in a company developing enterprise resource planning (ERP) software solutions. The company in question is one of the largest software companies in Türkiye, accompanying the sustainable growth journey of more than 200,000 companies. It offers its solutions developed specifically for sectors such as e-commerce, retail, finance, telecommunications, aviation and automotive. This study was conducted with a senior business analyst and analysis and testing team leader in a software development (SD) team under digital transformation services in a software company. In the examined SD department, solutions that can be customized according to needs are offered in the areas of treasury management, business-to-business (B2B) systems, data collection, consolidation, and central reporting.

4.1.Preliminaries

The criteria pool obtained as a result of the literature review was consolidated by taking expert opinion and the criteria list in Table 5 was obtained. The criteria are primarily grouped under 4 main headings: cost, quality, time, and scope. Later, they were divided into subgroups within themselves.

Table 5. Criteria used in productivity measurement

Category	Criteria Group	Criterion
Cost	Process Costs	Scoping Cost
		Support Cost
		Analysis Cost
		Testing Cost
		Deployment Cost
		Software Development Cost
Quality	Software Quality	Flexibility
		Extensibility
		Reliability
		Code Quality
		Usability
		Ease of Testing
		Operability
	Analysis Design Quality	Integrity
		Correctness of Design
		Ease of Use
Time	Delivery Times	Scoping Duration
		Support Duration
		Analysis Duration
		Test Duration
		Deployment Duration
		Software Development Duration
	Time Planning	Delivery of Priority Works
Scope	External factors	Communication with the Customer
		Compliance with the Customer
		Adaptation to Change in Project Needs
		Adaptation to Technological Developments
	Internal Factors	Document Requirement
		Number of Documents
		Project Complexity
		Customer Satisfaction
		Fulfilling Customer Demand Correctly
		Customer Involvement in the Process
		Meeting the Customer's Additional Demands
		Customer Involvement in the Process
		Adaptation to Organizational Habits
		Planning
	Team Status	Organized Structure
		Efficiency of Meetings
		Concentration of Team Members
		Motivation of Team Members
		Team Harmony

The linguistic term set in Table 6 was determined to receive qualitative expert assessments. The terms are designated as triangular FS. The symbols in the table mean: l - left support point, m - core, r – right support point, and d – arithmetic average of l, m, and r. In this notation, core refers to the point having maximum membership degree, left support point refers to the lower limit and right support point refers to the upper limit for the members of the fuzzy set. These linguistic terms can be symbolized as shown in Figure 5.

Table 6. Linguistic term set and fuzzy set parameters

Evaluation Scale	Abbreviation	Fuzzy Set Parameters **l**- Left Support Point	**m** - Core	**r** – Right Support Point	**d** – Arithmetic Average
None	N	0	0	0	0
Very Low	VL	0	0	0.3	0.1
Low	L	0	0.25	0.5	0.25
Medium	M	0.3	0.5	0.7	0.5
High	H	0.5	0.75	1	0.75
Very High	VH	0.7	1	1	0.9
Perfect	P	1	1	1	1

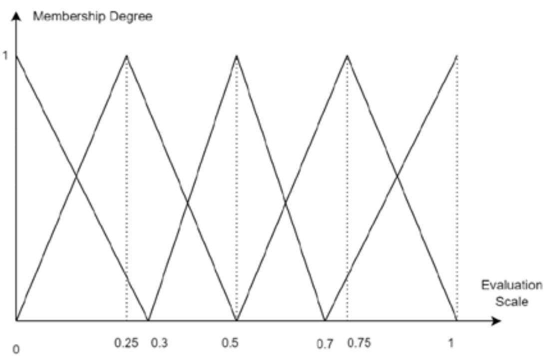

Fig. 5. Linguistic Terms and Associated Fuzzy Sets

4.2.Data collecting

Data collection from experts was carried out in two stages. In the first stage, the experts were asked to rank the criteria in order of importance and to determine their relative importance to each other using the linguistic terms in Table 5 with the help of the linguistic term set in Table 5. Experts can make this evaluation in the form of a 2-tuple LFM. These evaluations provide the appropriate input for SWARA calculations.

As a second step, the experts were asked to evaluate the agile and waterfall approaches for qualitative and quantitative criteria. For qualitative evaluations, they were asked to use the linguistic terms in Table 5 in the form of a 2-tuple LFM. The expert evaluations made are given in Appendix 1.

4.3.Data Processing and Calculation

After evaluating the qualitative and quantitative criteria, the data were converted into numerical expressions and rescaled using the methods described in Section 3.2.3. Then, productivity values were calculated as described in Section 3.2.4. The obtained numerical evaluations and calculation steps are given in Appendix 2.

4.4.Application Results

Productivity values and productivity comparisons obtained at 100%, 99.9% and 99% confidence intervals, respectively, are given in Table 7.

To visualize the data in Table 7, comparisons of methodologies for productivity at each confidence interval are given in Figure 6.

Table 7. Comparisons of productivity calculations in confidence intervals

Metric	**Ratio (100% Confidence)**	**Ratio (99.9% Confidence)**	**Ratio (99% Confidence)**
Productivity of Agile Methodology (Compared to Ideal Scenario)	69.19	69.21	71.07
Productivity of Waterfall Model (Compared to Ideal Scenario)	42.38	42.34	43.29
Productivity Increase of Agile Methodology (Compared to Ideal Scenario)	26.82	26.87	27.78
Productivity Increase of Agile Methodology (Compared to Waterfall)	63.29	63.46	64.18

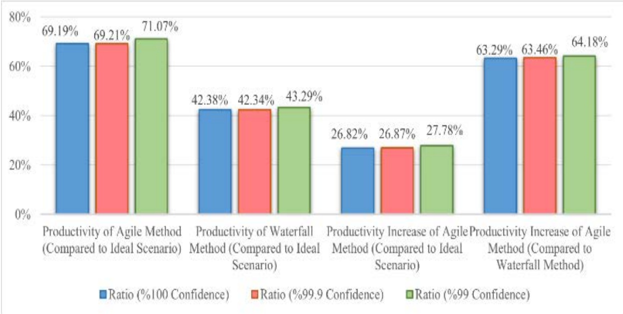

Fig. 6. Percentage productivity comparisons in confidence intervals

When Table 7 is examined, the productivity of agile methodologies compared to the ideal was calculated as 69.19% in 100% confidence interval, 69.21% in 99.9% confidence interval, and 71.07% in 99% confidence interval. The productivity of the waterfall model was calculated as 42.38% in 100% confidence interval, 42.34% in 99.9% confidence interval, and 43.29% in 99% confidence interval based on the ideal scenario. With the transition from waterfall model to agile methodology, compared to the ideal scenario, a productivity increase of 26.82 in the 100% confidence interval, 26.87 in the 99.9% confidence interval, and 27.78 in the 99% confidence interval was observed. When the findings are converted into percentages, it can be said that agile methodologies are more efficient than the waterfall model with the values of 63.29% in the 100% confidence interval, 63.46% in the 99.9% confidence interval, and 64.18% in the 99% confidence interval. Although fewer criteria were used in the 99% confidence interval compared to the calculation made for the 100% confidence interval, no significant difference was observed between the productivity values. Thus, it can be concluded that 19 of the criteria are not very effective in calculating productivity. These criteria are: analysis cost, support cost, analysis design quality - ease of use, software quality - code quality, compliance with organizational habits, test cost, delivery times - test time, software quality - test ease, compliance with technological developments, document need, team productivity of meetings, delivery times-dissemination time, scope cost, organizational structure within the team, concentration of team members, motivation of team members, team cohesion, deployment cost. number of documents.

Among the criteria that have a high impact on productivity, those related to the customer are at the forefront. This shows us that in agile methodologies, the project is divided into phases and the customer is involved in every stage, thus increasing its impact on productivity. It should be noted that in software PM, the priority is the customers. Then comes the delivery times of priority works. Here, while the project is being planned, priority works should be determined, and appropriate times should be given. The accuracy of the analysis should be high so that there is no disruption during the project. In the scope of work to be done, the needs should be determined and should not be kept too long. The analysis period should be planned in a way that satisfies the employee and the customer. If it is kept shorter than the ideal, it will stress the employee and lead him to error. If it takes too long, it both increases the cost for the customer and affects satisfaction negatively. In terms of software quality, ease of use, smooth operation, cost, extensibility, and flexibility of the software are other factors that increase productivity.

5. Conclusions

Although software projects have unique features, they have basic features in terms of PM. However, unlike other types of projects, the most distinctive feature of software projects is that they focus on abstract intellectual processes rather than physical products. Software projects require special management techniques, as management processes of software projects may encounter difficulties and uncertainties due to their abstract nature. The agile project management approach has become quite common in recent years because it provides more efficient management in situations where uncertainty is high. However, the discourse about the increase in productivity is generally not supported by numerical values and is expressed at the slogan level. Although comparisons involving verbal evaluations are made in the literature, no approach provides a comparative numerical measurement.

In this study, a model for measuring the productivity of software projects and a linguistic fuzzy SWARA-based procedure that enables this model to be operated are proposed. Using this procedure, the productivity of agile methodologies can be scaled compared to the ideal scenario and waterfall model, and the productivity increase compared to the waterfall model can be calculated. The proposed procedure was applied with the help of expert opinions in a company that offers ERP software solutions, and it was concluded that the agile approach provided a 63% increase in productivity compared to the waterfall model in the examined business.

Although the obtained result shows that the agile approach is much more efficient than the waterfall, it should be underlined that this is not a general result covering all software projects. For each project, the methodology to be applied should be decided by considering the dynamics of the project. In today's rapidly developing technological environment, all business processes, market dynamics and customer requirements are constantly changing. In this dynamic environment, agile methodologies may seem more advantageous compared to the waterfall model, but in some cases, they may not provide the structure and stability provided by the waterfall model. Therefore, a successful software project manager must comprehensively understand the advantages and disadvantages of agile and waterfall approaches and choose the most appropriate methodology based on the project's requirements.

In future studies, the number of experts and criteria for evaluation should be increased and people from different sectors should be reached. Thus, it is expected that more accurate results will emerge in the comparisons to be made.

Declaration of conflicting interests

The authors declare no competing interests.

Funding

The author received no financial support for the research and/or authorship of this article.

References

[1] Adebayo, A. and Lassi, L., "The relationship between traditional project management, agile project management and teamwork quality on project success", *International Journal of Organizational Analysis*, 30(7): 124-136, 2022.

[2] Gencer, C. and Kayacan, A., "Yazılım proje yönetimi: şelale modeli ve çevik yöntemlerin karşılaştırılması", *Bilişim Teknolojileri*: 33-75, 2017.

[3] Pmbok "A guide to the project management body of knowledge, sixth edition" *Newtown Square*, PA: Project,2007.

[4] Yıldız, D., "Bilişim sistemi yazilim geliştirme sürecinde kullanilan şelale ve scrum yöntemlerinin karşilaştirilmasi ve karma yöntemin incelenmesi", *Yönetim Bilişim Sistemleri*, 7(1): 13-20, 2021.

[5] Westland, J., *"The project management life cycle: A complete step-by-step methodology for initiating planning executing and closing the project:" Kogan Page Publishers*,2007.

[6] Lock, D., "Project management:" Routledge,2020.

[7] Ciric, D., Lalic, B., Gracanin, Nemanja, T., Milan, D., and Nenad, M., "Agile vs. Traditional Approach in Project Management: Strategies, Challenges and Reasons to Introduce Agile. 25th International "*Conference on Production Research Manufacturing Innovation: Cyber Physical Manufacturing*, Chicago, Illinois (USA),2019.

[8] Elibol, M. and Erol, Ç.S., "Scrum metodu kullanılarak bir mobil uygulama geliştirme sürecinin gerçekleştirilmesi", *Bilişim Teknolojileri*, 10(2): 170-171, 2017.

[9] Agile Turkey.9th Annual Agility Report Of Turkey, 2020.

[10] Schwaber, K., "Scrum development process. Business Object Design and Implementation: "*OOPSLA'95 Workshop Proceedings, Austin, Texas: Springer London*,1997.

[11] Dobrigkeit, F., Matthies, C., Teusner, R., and Perscheid, M., "Joining forces: Applying design thinking techniques in scrum meetings", *Design Thinking Research: Translation, Prototyping, and Measurement*: 315-345, 2021.

[12] Atabey, S. Scrum. Retrieved from https://www.komhedos.com/scrum/, 2019

[13] Standish Group International. CHAOS Report 2015. Retrieved from https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf

[14] Macit, Y. and Tüzün, E.,*"Uygulama Yaşam Döngüsü Yönetimi Karşılaştırmalı Süreç İncelemesi." 9'uncu Ulusal Yazılım Mühendisliği Sempozyumu-UYMS*,2018.

[15] Macit, Y. and Tüzün, E., "Uygulama Yaşam Döngüsü Yönetimi Karşılaştırmalı Süreç İncelemesi." *9'uncu Ulusal Yazılım Mühendisliği Sempozyumu-UYMS*,2015.

[16] Zavyalova, E., Sokolov, D., and Lisosvkaya, A., "Agile vs traditional project management approaches: Comparing human resource management architectures", *International Journal of Organizational Analysis*, 30(5): 1095-1112, 2020.

[17] Bojan, G., "Traditional vs agile project management in the service sector", *European Project Management Journal*, 2(9): 55-63, 2019.

[18] Kaim, R., Härting, R.C., and Reichstein, C., 2019, *Benefits of Agile Project Management in an Environment of Increasing Complexity—A Transaction Cost Analysis*. Proceedings of the 11th KES International Conference on Intelligent Decision Technologies.

[19] Knut, L., "Traditional and Agile Management Approaches. "*12th ILERA European Congress*, Düsseldorf, 2019.

[20] Rabia, M.S., Sayed, A.S., and Muhammad Tuaha, H.H., "A Review of Agile Methodology in IT Projects." *Proceedings of 2nd International Conference on Advanced Computing and Software Engineering*,2019

[21] Daniel, R. and Amy, C., "An agile framework for teaching with scrum in the it project management classroom", *Journal of Information Systems Education*, 31(3): 196-207, 2020.

[22] Abdallah, L., Gabriela, F., and Sassi, S.B., "A conceptual hybrid project management model for construction projects", *Procedia Computer Science*, 181: 921-930, 2021.

[23] Sanchez, F., Micaelli, J.P., and Bonjour, E., "A step for improving the transition between traditional project management to agile project management using a project management maturity model", *The Journal of Modern Project Management*, 7(1), 2022.

[24] Pace, M., "A correlational study on project management methodology and project success", *Journal of Engineering, Project, and Production Management*, 9(2): 56-65, 2019.

[25] Dursun, M. and Goker , N., "Evaluation of project management methodologies success factors using fuzzy cognitive map method: waterfall, agile, and lean six sigma cases", *International Journal of Intelligent Systems And Applications In Engineering*, 10(1): 35-43, 2021.

[26] Eren, N.L., "Proje yönetiminde dijital dönüşüm süreçleri için bir bulanık çkkv modeli önerisi (MSc. Thesis), 2022.

[27] Keršuliene, V., Zavadskas, E.K., and Turskis, Z., "Selection of rational dispute resolution method by applying new step-wise weight assessment ratio analysis (SWARA)", *Journal of Business Economics and Management*, 11(2): 243-258, 2010.

[28] Şengül, D., Çağıl, G., and Ardalı, Z., "Bulanık swara ve aralık değerli sezgisel bulanık ahp yöntemi ile iş değerlemesi", *Yönetim ve*

Ekonomi, 28(2): 248-250, 2021.

[29] Wang, J., Xu, L., Cai, J., and Fu, Y., "Offshore wind turbine selection with a novel multi-criteria decision-making method based on Dempster-Shafer evidence theory", *Sustainable Energy Technologies and Assessments*, 51 51, 101951), 2022.

[30] Işık, G., "2-öğeli tam dilsel sezgisel bulanık küme yaklaşımı: karar verme üzerine bir uygulama", *Gazi University Journal of Science Part C: Design and Technology*, 10(4): 826-840, 2022.

Apendix 1: Expert Evaluations

Table 8. Evaluations of experts

Criteria Code	Criteria Name	Criteria Type	Agile	Waterfall
K1	Scoping Cost	Cost	7	10
K2	Support Cost	Cost	20	26
K3	Analysis Cost	Cost	35	45
K4	Testing Cost	Cost	20	20
K5	Deployment Cost	Cost	5	4
K6	Software Development Cost	Cost	80	105
K7	Flexibility	Benefit	VH, +0.3	M, -0.2
K8	Extensibility	Benefit	VH, +0.1	L, -0.3
K9	Reliability	Benefit	VH	M
K10	Code Quality	Benefit	H	M
K11	Usability	Benefit	VH	M
K12	Ease of Testing	Benefit	VH	M
K13	Operability	Benefit	VH	M
K14	Integrity	Benefit	H	VH
K15	Correctness of Design	Benefit	VH	H
K16	Ease of Use	Benefit	H	H
K17	Scoping Duration	Cost	10	15
K18	Support Duration	Cost	15	20
K19	Analysis Duration	Cost	40	51
K20	Test Duration	Cost	25	25
K21	Deployment Duration	Cost	5	4
K22	Software Development Duration	Cost	88	120
K23	Delivery Times of Priority Works	Cost	VH	L
K24	Communication with the Customer	Benefit	VH	M
K25	Compliance with the Customer	Benefit	H	M
K26	Adaptation to Change in Project Needs	Benefit	VH	L
K27	Adaptation to Technological Developments	Benefit	M	L
K28	Document Requirement	Cost	H	H
K29	Number of Documents	Cost	M	VH
K30	Project Complexity	Cost	VH	M
K31	Customer Satisfaction	Benefit	H	M
K32	Fulfilling Customer Demand Correctly	Benefit	VH	H
K33	Meeting the Customer's Additional Demands	Benefit	VH	M
K34	Customer Involvement in the Process	Benefit	VH, +0.3	H, -0.4
K35	Compliance with Organizational Habits	Benefit	H	M
K36	Planning	Benefit	H	L
K37	Organized Structure	Benefit	H	M
K38	Efficiency of Meetings	Benefit	P, -0.2	M
K39	Concentration of Team Members	Benefit	VH	M
K40	Motivation of Team Members	Benefit	VH	M
K41	Team Harmony	Benefit	VH	H

IJPTE Vol, 02, No.02, pp. 147-164 December. 2023

Table 9. Prioritization of criteria assessed by experts

Criteria Code	Criteria Name	Order of Importance	Importance Level According to the Previous
K32	Fulfilling Customer Demand Correctly	1	-
K24	Communication with the Customer	2	L
K34	Customer Involvement in the Process	3	VL
K23	Delivery Times of Priority Works	4	VL
K15	Correctness of Design	5	N
K17	Scoping Duration	6	L
K19	Analysis Duration	7	VL
K11	Usability	8	L
K26	Adaptation to Change in Project Needs	9	VL
K14	Integrity	10	N
K31	Customer Satisfaction	11	VL
K13	Operability	12	N
K22	Software Development Duration	13	M
K6	Software Development Cost	14	VL
K8	Extensibility	15	L
K7	Flexibility	16	N
K36	Planning	17	VL
K9	Reliability	18	N
K25	Compliance with the Customer	19	N
K18	Support Duration	20	N
K30	Project Complexity	21	L
K33	Meeting the Customer's Additional Demands	22	VL
K3	Analysis Cost	23	L
K2	Support Cost	24	VL
K16	Ease of Use	25	L
K10	Code Quality	26	VL
K35	Compliance with Organizational Habits	27	VL
K4	Testing Cost	28	L
K20	Test Duration	29	N
K12	Ease of Testing	30	N
K27	Adaptation to Technological Developments	31	M
K28	Document Requirement	32	VL
K38	Efficiency of Meetings	33	N
K21	Deployment Duration	34	VL
K1	Scoping Cost	35	VL
K37	Organized Structure	36	VL
K39	Concentration of Members	37	N
K40	Motivation of Team Members	38	N
K41	Team Harmony	39	N
K5	Deployment Cost	40	L
K29	Number of Documents	41	VH

Appendix 2: Calculation Steps of the Proposed Methodology for Different Confidence Intervals

Table 10. The numerical equivalent of expert evaluations

Criteria Code	Criteria Name	Criteria Type	Agile	Waterfall
K1	Scoping Cost	Cost	7	10
K2	Support Cost	Cost	20	26
K3	Analysis Cost	Cost	35	45
K4	Testing Cost	Cost	20	20
K5	Deployment Cost	Cost	5	4
K6	Software Development Cost	Cost	80	105
K7	Flexibility	Benefit	5.3	2.8
K8	Extensibility	Benefit	5.1	1.7
K9	Reliability	Benefit	5	3
K10	Code Quality	Benefit	4	3
K11	Usability	Benefit	5	3
K12	Ease of Testing	Benefit	5	3
K13	Operability	Benefit	5	3
K14	Integrity	Benefit	4	5
K15	Correctness of Design	Benefit	5	4
K16	Ease of Use	Benefit	4	4
K17	Coverage Period	Cost	10	15
K18	Support Period	Cost	15	20
K19	Analysis Time	Cost	40	51
K20	Test Period	Cost	25	25
K21	Deployment Time	Cost	5	4
K22	Software Development Duration	Cost	88	120
K23	Delivery Times of Priority Works	Cost	5	2
K24	Communication with the Customer	Benefit	5	3
K25	Compliance with the Customer	Benefit	4	3
K26	Adaptation to Change in Project Needs	Benefit	5	2
K27	Adaptation to Technological Developments	Benefit	3	2
K28	Document Requirement	Cost	4	4
K29	Number of Documents	Cost	3	5
K30	Project Complexity	Cost	5	3
K31	Customer Satisfaction	Benefit	4	3
K32	Fulfilling Customer Demand Correctly	Benefit	5	4
K33	Meeting the Customer's Additional Demands	Benefit	5	3
K34	Customer Involvement in the Process	Benefit	5.3	3.6
K35	Compliance with Organizational Habits	Benefit	4	3
K36	Planning	Benefit	4	2
K37	Organized Structure	Benefit	4	3
K38	Efficiency of Meetings	Benefit	5.8	3
K39	Concentration of Team Members	Benefit	5	3
K40	Motivation of Team Members	Benefit	5	3
K41	Team Harmony	Benefit	5	4

Table 11. Rescaled expert evaluations

Criteria Code	Criteria Name	Criteria Type	Agile	Waterfall
K1	Scoping Cost	Cost	0.3000	0.0000
K2	Support Cost	Cost	0.2308	0.0000
K3	Analysis Cost	Cost	0.2222	0.0000
K4	Testing Cost	Cost	0.0000	0.0000
K5	Deployment Cost	Cost	0.0000	0.2000
K6	Software Development Cost	Cost	0.2381	0.0000
K7	Flexibility	Benefit	0.8833	0.4667
K8	Extensibility	Benefit	0.8500	0.2833
K9	Reliability	Benefit	0.8333	0.5000
K10	Code Quality	Benefit	0.6667	0.5000
K11	Usability	Benefit	0.8333	0.5000
K12	Ease of Testing	Benefit	0.8333	0.5000
K13	Operability	Benefit	0.8333	0.5000
K14	Integrity	Benefit	0.6667	0.8333
K15	Correctness of Design	Benefit	0.8333	0.6667
K16	Ease of Use	Benefit	0.6667	0.6667
K17	Coverage Period	Cost	0.3333	0.0000
K18	Support Period	Cost	0.2500	0.0000
K19	Analysis Time	Cost	0.2157	0.0000
K20	Test Period	Cost	0.0000	0.0000
K21	Deployment Time	Cost	0.0000	0.2000
K22	Software Development Duration	Cost	0.2667	0.0000
K23	Delivery Times of Priority Works	Cost	0.8333	0.3333
K24	Communication with the Customer	Benefit	0.8333	0.5000
K25	Compliance with the Customer	Benefit	0.6667	0.5000
K26	Adaptation to Change in Project Needs	Benefit	0.8333	0.3333
K27	Adaptation to Technological Developments	Benefit	0.5000	0.3333
K28	Document Requirement	Cost	0.6667	0.6667
K29	Number of Documents	Cost	0.5000	0.8333
K30	Project Complexity	Cost	0.8333	0.5000
K31	Customer happiness	Benefit	0.6667	0.5000
K32	Fulfilling Customer Demand Correctly	Benefit	0.8333	0.6667
K33	Meeting the Customer's Additional Demands	Benefit	0.8333	0.5000
K34	Customer Involvement in the Process	Benefit	0.8833	0.6000
K35	Compliance with Organizational Habits	Benefit	0.6667	0.5000
K36	Planning	Benefit	0.6667	0.3333
K37	Organized Structure	Benefit	0.6667	0.5000
K38	Efficiency of Meetings	Benefit	0.9667	0.5000
K39	Concentration of Members	Benefit	0.8333	0.5000
K40	Members' Motivation	Benefit	0.8333	0.5000
K41	Team Cohesion	Benefit	0.8333	0.6667

Table 12. Criteria weights calculated by SWARA in the different confidence intervals

Criteria Code	s_j	k_j	q_j	w_j **100%**	$w_J/\min(w_j > 1\%)$ **99%**	$w_J/\min(w_j > 1\%)$ **99.9%**
K32	-	1	1	11.481%	12.40%	11.49%
K24	0.25	1.25	0.8	9.185%	9.92%	9.19%
K34	0.1	1.1	0.7273	8.350%	9.02%	8.36%
K23	0.1	1.1	0.6612	7.591%	8.20%	7.60%
K15	0	1	0.6612	7.591%	8.20%	7.60%
K17	0.25	1.25	0.5289	6.072%	6.56%	6.08%
K19	0.1	1.1	0.4808	5.520%	5.96%	5.53%
K11	0.25	1.25	0.3847	4.416%	4.77%	4.42%
K26	0.1	1.1	0.3497	4.015%	4.34%	4.02%
K14	0	1	0.3497	4.015%	4.34%	4.02%

K31	0.1	1.1	0.3179	3.650%	3.94%	3.65%
K13	0	1	0.3179	3.650%	3.94%	3.65%
K22	0.5	1.5	0.2119	2.433%	2.63%	2.44%
K6	0.1	1.1	0.1927	2,212%	2.39%	2.21%
K8	0.25	1.25	0.1541	1.770%	1.91%	1.77%
K7	0	1	0.1541	1.770%	1.91%	1.77%
K36	0.1	1.1	0.1401	1.609%	1.74%	1.61%
K9	0	1	0.1401	1.609%	1.74%	1.61%
K25	0	1	0.1401	1.609%	1.74%	1.61%
K18	0	1	0.1401	1.609%	1.74%	1.61%
K30	0.25	1.25	0.1121	1.287%	1.39%	1.29%
K33	0.1	1.1	0.1019	1.170%	1.26%	1.17%
K3	0.25	1.25	0.0815	0.936%		0.94%
K2	0.1	1.1	0.0741	0.851%		0.85%
K16	0.25	1.25	0.0593	0.681%		0.68%
K10	0.1	1.1	0.0539	0.619%		0.62%
K35	0.1	1.1	0.0490	0.563%		0.56%
K4	0.25	1.25	0.0392	0.450%		0.45%
K20	0	1	0.0392	0.450%		0.45%
K12	0	1	0.0392	0.450%		0.45%
K27	0.5	1.5	0.0261	0.300%		0.30%
K28	0.1	1.1	0.0238	0.273%		0.27%
K38	0	1	0.0238	0.273%		0.27%
K21	0.1	1.1	0.0216	0.248%		0.25%
K1	0.1	1.1	0.0196	0.225%		0.23%
K37	0.1	1.1	0.0179	0.205%		0.21%
K39	0	1	0.0179	0.205%		0.21%
K40	0	1	0.0179	0.205%		0.21%
K41	0	1	0.0179	0.205%		0.21%
K5	0.25	1.25	0.0143	0.164%		0.16%
K29	0.9	1.9	0.0075	0.086%		

Table 13. Productivity calculations at 100% confidence interval compared to the ideal scenario

Criteria Code	Criteria Type	Weight (100%)	Agile	Waterfall	Difference
K32	Benefit	0.1148	0.833	0.667	0.167
K24	Benefit	0.0918	0.833	0.500	0.333
K34	Benefit	0.0835	0.883	0.600	0.283
K23	Cost	0.0759	0.833	0.333	0.500
K15	Benefit	0.0759	0.833	0.667	0.167
K17	Cost	0.0607	0.333	0.000	0.333
K19	Cost	0.0552	0.216	0.000	0.216
K11	Benefit	0.0442	0.833	0.500	0.333
K26	Benefit	0.0401	0.833	0.333	0.500
K14	Benefit	0.0401	0.667	0.833	-0.167
K31	Benefit	0.0365	0.667	0.500	0.167
K13	Benefit	0.0365	0.833	0.500	0.333
K22	Cost	0.0243	0.267	0.000	0.267
K6	Cost	0.0221	0.238	0.000	0.238
K8	Benefit	0.0177	0.850	0.283	0.567
K7	Benefit	0.0177	0.883	0.467	0.417
K36	Benefit	0.0161	0.667	0.333	0.333
K9	Benefit	0.0161	0.833	0.500	0.333
K25	Benefit	0.0161	0.667	0.500	0.167
K18	Cost	0.0161	0.250	0.000	0.250
K30	Cost	0.0129	0.833	0.500	0.333
K33	Benefit	0.0117	0.833	0.500	0.333
K3	Cost	0.0094	0.222	0.000	0.222
K2	Cost	0.0085	0.231	0.000	0.231
K16	Benefit	0.0068	0.667	0.667	0.000

K10	Benefit	0.0062	0.667	0.500	0.167
K35	Benefit	0.0056	0.667	0.500	0.167
K4	Cost	0.0045	0.000	0.000	0.000
K20	Cost	0.0045	0.000	0.000	0.000
K12	Benefit	0.0045	0.833	0.500	0.333
K27	Benefit	0.0030	0.500	0.333	0.167
K28	Cost	0.0027	0.667	0.667	0.000
K38	Benefit	0.0027	0.967	0.500	0.467
K21	Cost	0.0025	0.000	0.200	-0.200
K1	Cost	0.0023	0.300	0.000	0.300
K37	Benefit	0.0020	0.667	0.500	0.167
K39	Benefit	0.0020	0.833	0.500	0.333
K40	Benefit	0.0020	0.833	0.500	0.333
K41	Benefit	0.0020	0.833	0.667	0.167
K5	Cost	0.0016	0.000	0.200	-0.200
K29	Cost	0.0009	0.500	0.833	-0.333

Table 14. Productivity calculations at the 99.9% confidence interval compared to the ideal scenario

Criteria Code	Criteria Type	Weight (99.9%)	Agile	Waterfall	Difference
K32	Benefit	0.1149	0.833	0.667	0.167
K24	Benefit	0.0919	0.833	0.500	0.333
K34	Benefit	0.0836	0.883	0.600	0.283
K23	Cost	0.0760	0.833	0.333	0.500
K15	Benefit	0.0760	0.833	0.667	0.167
K17	Cost	0.0608	0.333	0.000	0.333
K19	Cost	0.0553	0.216	0.000	0.216
K11	Benefit	0.0442	0.833	0.500	0.333
K26	Benefit	0.0402	0.833	0.333	0.500
K14	Benefit	0.0402	0.667	0.833	-0.167
K31	Benefit	0.0365	0.667	0.500	0.167
K13	Benefit	0.0365	0.833	0.500	0.333
K22	Cost	0.0244	0.267	0.000	0.267
K6	Cost	0.0221	0.238	0.000	0.238
K8	Benefit	0.0177	0.850	0.283	0.567
K7	Benefit	0.0177	0.883	0.467	0.417
K36	Benefit	0.0161	0.667	0.333	0.333
K9	Benefit	0.0161	0.833	0.500	0.333
K25	Benefit	0.0161	0.667	0.500	0.167
K18	Cost	0.0161	0.250	0.000	0.250
K30	Cost	0.0129	0.833	0.500	0.333
K33	Benefit	0.0117	0.833	0.500	0.333
K3	Cost	0.0094	0.222	0.000	0.222
K2	Cost	0.0085	0.231	0.000	0.231
K16	Benefit	0.0068	0.667	0.667	0.000
K10	Benefit	0.0062	0.667	0.500	0.167
K35	Benefit	0.0056	0.667	0.500	0.167
K4	Cost	0.0045	0.000	0.000	0.000
K20	Cost	0.0045	0.000	0.000	0.000
K12	Benefit	0.0045	0.833	0.500	0.333
K27	Benefit	0.0030	0.500	0.333	0.167
K28	Cost	0.0027	0.667	0.667	0.000
K38	Benefit	0.0027	0.967	0.500	0.467
K21	Cost	0.0025	0.000	0.200	-0.200
K1	Cost	0.0023	0.300	0.000	0.300
K37	Benefit	0.0021	0.667	0.500	0.167
K39	Benefit	0.0021	0.833	0.500	0.333
K40	Benefit	0.0021	0.833	0.500	0.333
K41	Benefit	0.0021	0.833	0.667	0.167
K5	Cost	0.0016	0.000	0.200	-0.200
K29	Cost	0	0.500	0.833	-0.333

Table 15. Productivity calculations at the 99% confidence interval

Criteria Code	Criteria Type	Weight (99.9%)	Agile	Waterfall	Difference
K32	Benefit	0.1240	0.833	0.667	0.167
K24	Benefit	0.0992	0.833	0.500	0.333
K34	Benefit	0.0902	0.883	0.600	0.283
K23	Cost	0.0820	0.833	0.333	0.500
K15	Benefit	0.0820	0.833	0.667	0.167
K17	Cost	0.0656	0.333	0.000	0.333
K19	Cost	0.0596	0.216	0.000	0.216
K11	Benefit	0.0477	0.833	0.500	0.333
K26	Benefit	0.0434	0.833	0.333	0.500
K14	Benefit	0.0434	0.667	0.833	-0.167
K31	Benefit	0.0394	0.667	0.500	0.167
K13	Benefit	0.0394	0.833	0.500	0.333
K22	Cost	0.0263	0.267	0.000	0.267
K6	Cost	0.0239	0.238	0.000	0.238
K8	Benefit	0.0191	0.850	0.283	0.567
K7	Benefit	0.0191	0.883	0.467	0.417
K36	Benefit	0.0174	0.667	0.333	0.333
K9	Benefit	0.0174	0.833	0.500	0.333
K25	Benefit	0.0174	0.667	0.500	0.167
K18	Cost	0.0174	0.250	0.000	0.250
K30	Cost	0.0139	0.833	0.500	0.333
K33	Benefit	0.0126	0.833	0.500	0.333
K3	Cost	0	0.222	0.000	0.222
K2	Cost	0	0.231	0.000	0.231
K16	Benefit	0	0.667	0.667	0.000
K10	Benefit	0	0.667	0.500	0.167
K35	Benefit	0	0.667	0.500	0.167
K4	Cost	0	0.000	0.000	0.000
K20	Cost	0	0.000	0.000	0.000
K12	Benefit	0	0.833	0.500	0.333
K27	Benefit	0	0.500	0.333	0.167
K28	Cost	0	0.667	0.667	0.000
K38	Benefit	0	0.967	0.500	0.467
K21	Cost	0	0.000	0.200	-0.200
K1	Cost	0	0.300	0.000	0.300
K37	Benefit	0	0.667	0.500	0.167
K39	Benefit	0	0.833	0.500	0.333
K40	Benefit	0	0.833	0.500	0.333
K41	Benefit	0	0.833	0.667	0.167
K5	Cost	0	0.000	0.200	-0.200
K29	Cost	0	0.500	0.833	-0.333