AJIT-e Academic Journal of Information Technology
10.5824/ajite.2023.03.002.x

2023 . Summer – Yaz . Volume – Cilt: 14 . Issue – Sayı: 54

# Analysis and Comparison of Waterfall Model and Agile Approach in Software Projects

**Aşkın DEMİRAĞ**, *Yeditepe University, Faculty of Computer and Information Sciences, Assoc. Prof., ademirag@yeditepe.edu.tr, 0000-0001-7868-0438*

**Elçin Nur DEMİRKOL ÖZTÜRK**, *Yeditepe University, Lecturer, elcin.ozturk@yeditepe.edu.tr, 0009-0005-5848-7311*

**Ceyda ÜNAL**, *Dokuz Eylül University, Research Assistant,* ceyda.unal@deu.edu.tr, *0000-0002-5503-8124*

***ABSTACT***

*Looking at the history of civilization from past to present, the concepts of project and project management are frequently encountered. Thanks to project management, businesses can work target-oriented, provide high motivation, facilitate internal control, and provide a significant increase in quality.*

*The concept of software, born with the development of technology, penetrates our lives more and more day by day. The globalizing world has brought together the concepts of software and project management and integrated them. Software projects require special management techniques because of their content. For this reason, new methods have emerged over time in the management of software projects.*

*In this study, the concept of the project, the development of project management from its birth over the years, project management in software projects, software development life cycles, and the Waterfall Model and Agile Approach, which are two methods used in software projects, are discussed. The perception of the Waterfall Model and Agile Approach was evaluated with the survey study created with the participation of 145 employees from the software industry. Thus, it is aimed to support managers to choose an effective and efficient method at the point of deciding which method to proceed in software project management.*

***Keywords*** **:** Project Management, Software Projects, Software Development Life Cycles, The Waterfall Model, Agile Approach

# Yazılım Projelerinde Şelale Modeli ve Çevik Yaklaşımın Analizi ve Karşılaştırılması

**ÖZ**

*Geçmişten günümüze medeniyet tarihine bakıldığında proje ve proje yönetimi kavramlarına sıklıkla rastlanmaktadır. Proje yönetimi sayesinde işletmeler hedef odaklı çalışabilir, yüksek motivasyon sağlayabilir, iç kontrolü kolaylaştırabilir ve önemli ölçüde kalite artışı sağlayabilir.*



**Araştırma Makalesi / Research Article**
**Date Received:** 23.07.2023
**Publishing Date:** 30.08.2023

AJIT-e Academic Journal of Information Tecnology
2023 Summer/Yaz – Cilt/Vol: 14 - Sayı/Issue: 54
10.5824/ajite.2023.03.002.x

*Teknolojinin gelişmesiyle doğan yazılım kavramı her geçen gün hayatımıza daha çok giriyor. Küreselleşen dünya, yazılım ve proje yönetimi kavramlarını bir araya getirmiş ve bütünleştirmiştir. Yazılım projeleri, içerikleri nedeniyle özel yönetim teknikleri gerektirir. Bu nedenle yazılım projelerinin yönetiminde zaman içinde yeni yöntemler ortaya çıkmıştır.*
*Bu çalışmada proje kavramı, proje yönetiminin doğuşundan yıllar içinde gelişimi, yazılım projelerinde proje yönetimi, yazılım yaşam döngüleri ve yazılım projelerinde kullanılan iki yöntem olan Şelale Modeli ve Çevik Yaklaşım ele alınmaktadır. Yazılım sektöründen 145 çalışanın katılımıyla oluşturulan anket çalışması ile Şelale Modeli ve Çevik Yaklaşım algısı değerlendirilmiştir. Böylece yöneticilerin yazılım proje yönetiminde hangi yöntemi izleyeceklerine karar verme noktasında etkin ve verimli bir yöntem seçmelerine destek olunması amaçlanmaktadır.*

*Anahtar Kelimeler* : Proje Yönetimi, Yazılım Projeleri, Yazıl Geliştirme Yaşam Döngüsü, Şelale Model, Çevik Yaklaşım

## INTRODUCTION

Today, many businesses or individuals are closely interested in project management. Until the 1980s, project management focused mainly on providing information to senior managers about the resources used. However, especially after the nineties, with the business environment becoming much more complex, project management requires much more in today's conditions. Thus, the role of project management within companies is increasing in the face of increasing needs.

The great progress that information technologies have shown in recent times has led to a remarkable increase in productivity and efficiency, making information technologies an indispensable part of the business world. Therefore, the interest in software projects has increased day by day, and software projects have begun to be accepted and implemented by the wider masses.

The project management system and project management standards developed by the Project Management Institute (PMI), which is the most competent organization in the world in project management, are based on to provide an overview of software project management and guide the way to be followed.

The production of software projects requires a series of engineering and project management processes. The characteristics of the software to be produced, such as the size, complexity, purpose of use, the structure of the organization and project team that will produce the software, and the resources that can be used are important in the methodology selection of these processes. This study aims to examine the waterfall model, which is the pioneer of traditional plan-based methods and all kinds of project management methodologies in software project management, and agile methods that emerged as an alternative to

situations where traditional plan-based methods are insufficient and have gained a very important place today and are suitable for project-specific situations. The aim is to present a comparison that will guide the choice of methodology.

In the second part of the study, the concept of project management, in the third part, software project management concepts and management processes are discussed together with their knowledge areas. In the fourth chapter, the software life cycle, a brief history of the waterfall model and agile methods, roles and responsibilities, principles and features, benefits, and constraints are explained respectively. In the fifth chapter, the literature research conducted in the other chapters was supported by a questionnaire answered by 145 people who took part in agile projects from the software industry.

## 1. LITERATURE REVIEW

Before talking about the features and differences of the Waterfall method and Agile approach, it is necessary to talk about the concepts of project and project management. After these concepts, the concepts of software and software management were elaborated, and the analysis and comparison of the Waterfall method and Agile approach in software projects was provided.

According to Project Management Institute (PMBOK, 2017), "a project is a temporary endeavor undertaken to create a unique product, service, or result." The term "temporary" in this definition means that the project has a certain beginning and an inevitable end. In contrast, the term "unique" means that the product or service must be different from other products or services. (PMBOK, 2000, p. 4)

Project management is defined by Kerzner (2017) as a whole of planning, management, organizing, and oversight activities for the usage of resources, reducing costs and risks, maximizing the value while a project is executed for a business goal.

Project management is software from the information, technology, and tools used to meet the project requirements for the project. Project management brings together the appropriate version and application and the project management processes necessary for the project. Thanks to project management, it is ensured that organizations carry out projects effectively and efficiently. (Kerzner, 2017, p. 9)

According to PMBOK (2017), *"A Project Management Process Group is a logical grouping of project management processes to achieve specific project objectives. Process Groups are independent of project phases."*

Projects usually go through the same stages, from inception to completion. In the first phase of the life cycle that begins with implementing the project, a manager is appointed to the project and the key players to participate in the project, in other words, the team members. Then, the main objectives of the project to be outlined in the project mission are defined. At

this stage, the duties of the team leader, managers, and other staff in the team are defined. Project management processes are divided into five project management process groups: initiating, planning, executing, controlling, and closing. (Cleland et al., 2006, p. 1-17).

**Table 1:** Description of the Project Management Process Groups

| Process Group | Description | Common Terms |
|---|---|---|
| Initiating | Authorizing the project or phase. | "preliminary planning" "kicking off" |
| Planning | Defining and refining objectives of the "defining" project and selecting the best course of "developing the plan" action to attain those objectives. | "defining" "developing the plan." "setting the stage." |
| Executing | Coordinating the people and resources to "make it happen" implement the plan. | "making it happen." "getting it done." "coordinating" |
| Controlling | Ensuring project objectives are met by "tracking progress," monitoring and measuring progress regularly "keeping on course" to identify variances from the plan so that corrective actions can be taken. | "tracking progress" "keeping on course." |
| Closing | Formalizing acceptance of project or phase "client acceptance" and bringing to an orderly end. | "clientacceptance," "transition" "closeout |

According to ISO definitions, the software is programs, procedures, and documents that depend on the operation of a data processing system. Therefore, the software is an abstract product. The software is based on the plurality of logical paths in the program modules and the combination of all combinations of interface details.

Creating software requires innovative problem solving to create unique solutions. Software is a cognitive and human-based development process that requires the sharing of documents (Ruhe et al., 2014, p. 11 ).

Software projects are specialized projects, and while they have the basic features of project management, they require special management techniques due to their content. There are significant differences that distinguish software projects from other projects. While the final product's physical existence is created in other project management areas, the product

that occurs in software projects is abstract due to the intellectual processes. Therefore, it is more difficult to calculate the workforce and cost to be consumed in software projects, and the possibility of failure is higher since uncertainties are higher than the general project management. Projects carried out in other project management areas do not differ much according to time and place. The experience is gained in project management for these projects, and these experiences are used in new projects. In software projects, the rapid development of information technologies and changes in customer requirements may cause changes even in the current project and almost invalidate the experience gained in previous projects. For these reasons, success rates in software projects are low. The criteria determining whether projects are successful are time-resource-scope items, also known as the project management triangle. (Sommerville, 2011, p. 94)

In order to do all software development activities, different development models have been developed over time. While each model can eliminate a deficiency encountered in the previous model, different methodologies can still be used according to the size of the projects and the products to be produced, the structure of the customer and the business area, and a single model is not accepted as a standard.

All stages that the software product goes through during both production and customer use are called the software development life cycle. The software development process consists of stages based on timing and divided into the content. Software development life cycle (SDLC) covers software processes that transform software requirements into software products. Software product lifecycle (SPLC) includes software configuration management and software quality assurance processes in addition to a software development lifecycle and "end-to-end" processes such as commissioning, maintenance, support, evaluation, and retirement of software. There is no temporal sequence between software development processes individually. Software lifecycle models (SDLC or SPLC) create the sequences between these processes. Life cycle models define critical processes according to their philosophies and determine the sequences and interactions between these processes. (SWEBOK, p. 8-4).

The waterfall model is the oldest and basic model of software engineering in which the activities applied in software projects are performed in successive stages. This approach consists of completing the parts sequentially to form a whole. It is argued that one should not move on to the next one before completing a step. (Munassar et al.,2010, p. 95)

The customer provides all requirements at the beginning of the project. This method, which is suitable for projects with all requirements determined in detail, is unsuitable for projects that are likely to change requirements during project development. (Pressman et al., 2019, p. 25). In the waterfall approach, the entire software development process is divided into separate stages. In the waterfall model, typically, the result of one phase, in turn, provides input for the next phase. It consists of requirements analysis, design, coding and unit tests,

10.5824/ajite.2023.03.002.x

integration tests, and maintenance - repair stages. (Sommerville, 2007, p. 94). In the waterfall picture below, each step would be completed before moving on to the next step (Stanley et al., 2020, p. 21):

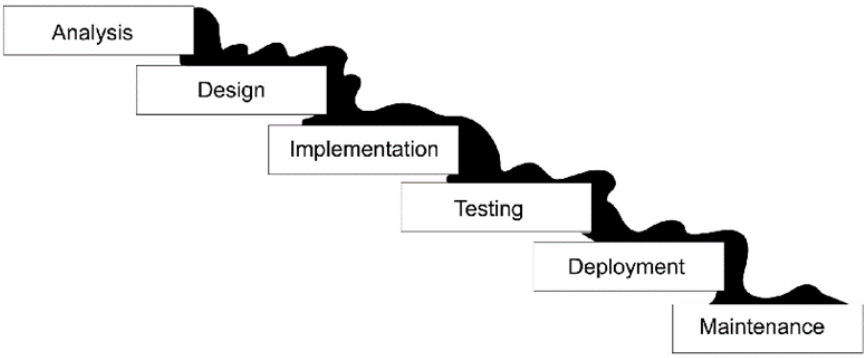


**Figure 1:** Waterfall Model Steps

Stanley et al.(2020) stated that "agile is a method of project management that focuses on dividing tasks into short phases of work, with frequent reviews of the project and adaption of planning mid-execution as needed."

Agile software development is a cost-effective iterative and incremental (evolutionary) approach to the changing needs of stakeholders. It meets with a highly collaborative approach within a practical governance framework that includes adequate formalization by teams that self-organize and produce high-quality solutions. As a result, agile methods have emerged as alternative solutions for issues where traditional methods are considered to be insufficient (Stanley et al. 2020, p. 41).

Project work is carried out continuously throughout the project period. Agile methods include adaptive planning, continuous improvement, and early delivery. It encourages agile, rapid, and flexible responses to change.

Iterations implemented in agile methods are formed by the experiences gained in previous iterations and the detected flaws. Assignments that require to be completed are prioritized according to the business evaluation they have. The project team decides how best job to do the work based on available resources and constraints. The project team must discharge determined tasks within a specified iteration time. It is the team, with its strengths and weaknesses, that is chargeable for forming the product to be delivered at the termination of the iteration. For this reason, in-team collaboration is essential. In this context, the basic principles on which agile methods are based can be listed as experimentality, prioritization, self-organization, time frame, cooperation.

This method is becoming increasingly popular and is often used to manage the software creation process. It focuses on teamwork and finishes one feature of the software at once (exactly) before moving on to the next.Agile iterative flow is seen in the figure below.(Stanley et al., 2020, p. 42).

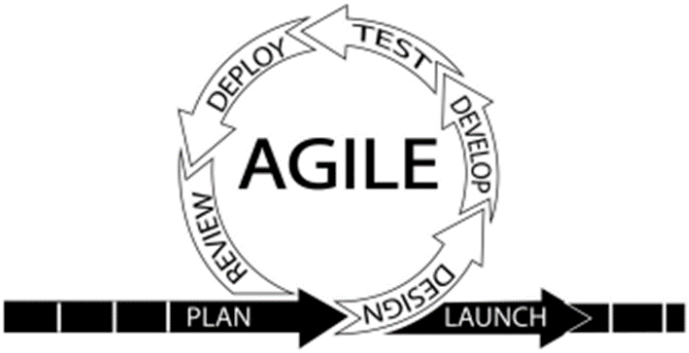


**Figure 2:** Agile Iteration Flow

## 2. MATERIAL-METHOD

In this study, it has been tried to determine how Agile and Waterfall software development methods are perceived by software developers about agile approaches in software organizations in Turkey. The research was carried out as a descriptive study based on a questionnaire. A survey was conducted with people working with Agile methods in the software industry. The purpose of descriptive analysis is to determine how much of the sample agrees with a certain opinion or how often certain events occur, rather than determining the relationships between variables. In this context, a 33-item questionnaire about software development methods was prepared. The questionnaire consists of two main parts. The first part was prepared to obtain demographic information about the participants. There are 12 questions in this section. In addition to demographic information, their total work experience, their experience in information technology, their positions and company sectors, the agile process models they use, the number of projects they have developed with agile methods were also learned. In the second part, they were asked to answer 21 questions focused on Agile Software development. The answers to the questions in this section have been prepared according to a 5-point Likert scale. The prepared questions have been previously designed on the subject, and this subject has been forwarded by the trainers for review and updates have been made in line with the feedback they have given. Thus, the surface and content validity of the study was ensured. All of the survey questions can be accessed online. The prepared survey was made available to the participants for three months during February, March and April 2020, using an online survey service called Google Forms. Participants were invited by using various social networks and e-mail groups. Many different institutions in the IT sector were reached and the opinions of 145 software developers were received.

10.5824/ajite.2023.03.002.x

## 3. FINDINGS AND DISCUSSION

Looking at the demographic results of the questionnaire , 63 percent of the participants are men and 37 percent are women.

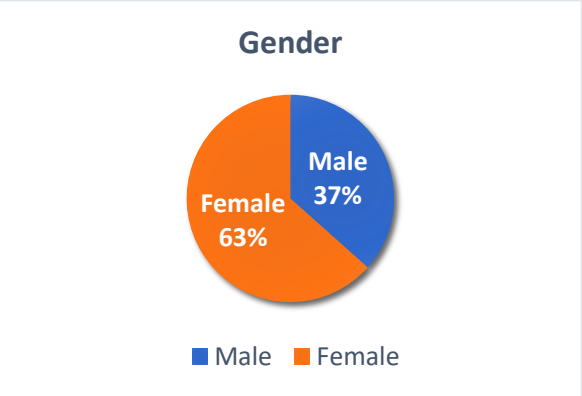


**Figure 3:** Distribution of Participants by Gender

When their distribution according to their work experience is examined; 23.4% had a total work experience of 0-5 years, 44.8% had 6-10 years, 23.4% had 11-15 years, 8.3% had 16 years or more, 29% 0-5 years of work experience in the sector, 42.1% of them 6-10 years, 21.4% of them 11-15 years, 6.9% of them 16 years and above, 72.4% It was determined that work experience in the institution was 0-5 years, 17.9% had 6-10 years, and 9.7% had 11 years or more.

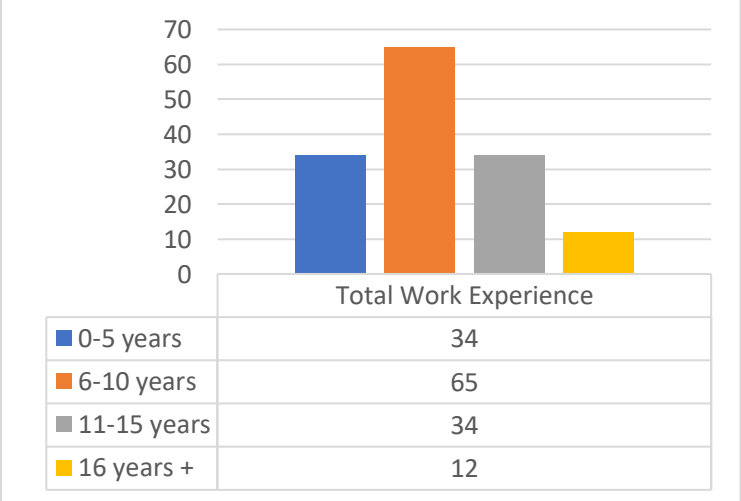


**Figure 4:** Distribution of Participants by Total Work Experience

29.0% of the participants are business/system analysts, 42.1% are software development specialists, 11.7% are managers, 17.2% are in other positions, 20.0% are in the banking/finance sector, It has been determined that 38.6% of them work in the airline/aviation sector, 10.3% in the production sector and 31.0% in other sectors.

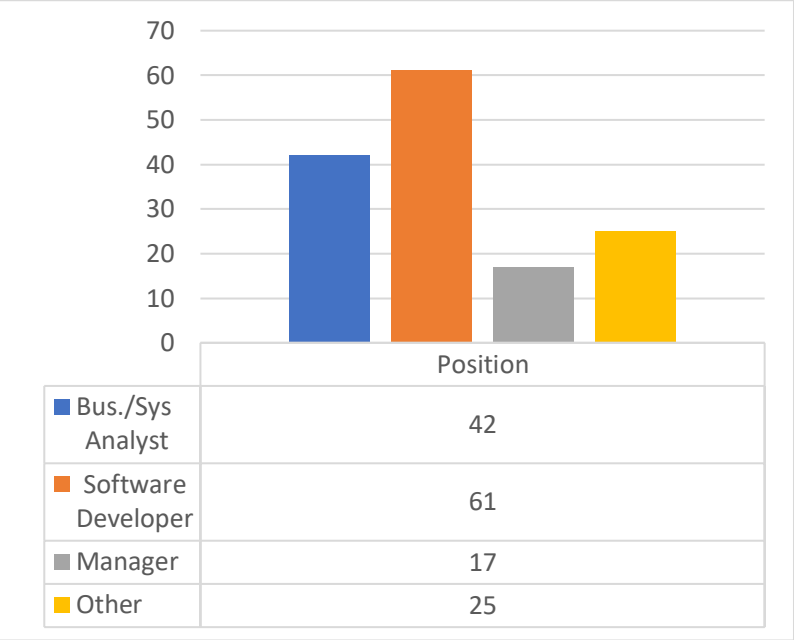


**Figure 5:** Distribution of Participants by Position

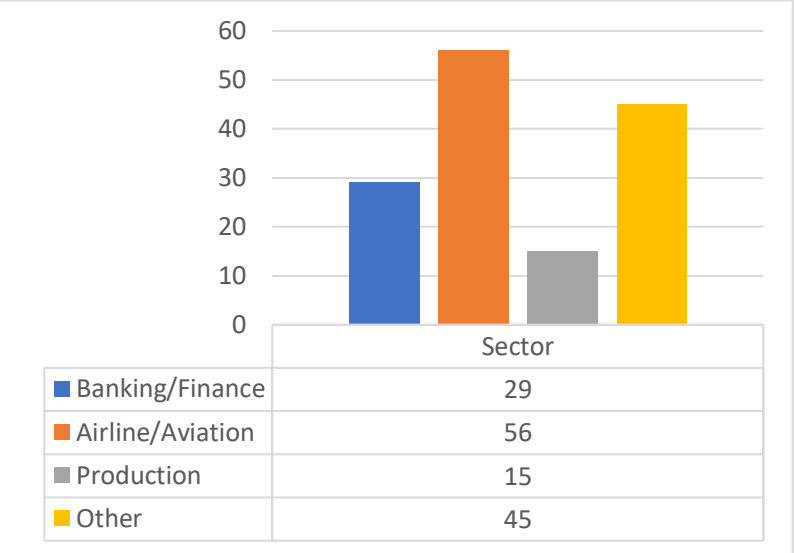


**Figure 6:** Distribution of Participants by Sector

When the distribution of the participants according to the average duration of the project is examined; When the distribution according to the number of people working in the average software project is examined, 37.9% of them are 1-3 months, 31.0% are 4-6 months, 31.0% are 7 months and more, and 47.6% are 1- When the distribution of 7 people, 20.7% 8-15 people, 31.7% 16 people or more, and the number of people working in the largest software project is analyzed; 1-10 people of 40.0%, 11-20 people of 24.1%, 21-30 people of 22.1%, 31 people and more of 13.8%, according to the number of people in the software project team When their distribution is examined; It was determined that 42.8% of them were 7 people or less, 57.2% were over 7 people.

10.5824/ajite.2023.03.002.x

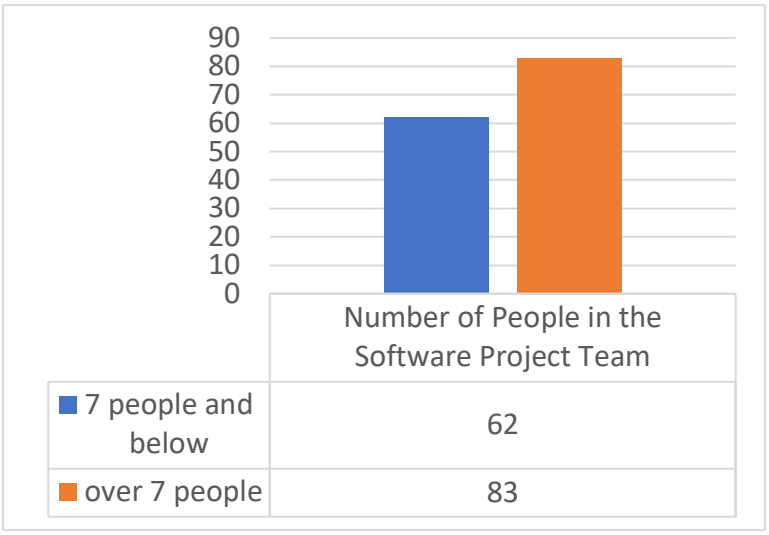


**Figure 7:** Distribution of Participants by Number of People in the Software Project Team

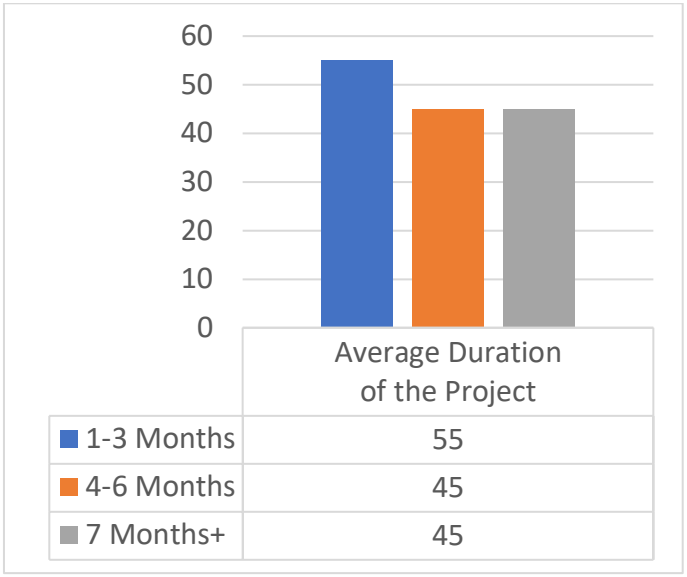


**Figure 8:** Distribution of Average Duration of the Project

In this part of the research, the descriptive statistics of the waterfall model and agile approach in the software projects of the participants and the findings of the determination of the differences according to the demographic characteristics are included. For this reason 6 sub-dimensions are divided.

In determining the distribution of the data obtained, arithmetic means, median, skewness, and kurtosis coefficients from central tendency measurements were used. It was determined that the distribution of the data obtained came from the normal distribution because the median and arithmetic mean values were close or equal to each other and the skewness and kurtosis values fell within ±2 limits.

**Table 2:** Normality Test Results of the Waterfall Model and Agile Approach in Software Projects of the Participants

| Scale and Sub-Dimensions | $\bar{X}$ | s.s. |
|---|---|---|
| **Teamwork and Motivation** | 3,78 | 0,49 |
| **Project Requirements** | 3,78 | 0,99 |
| **Efficiency and Quality** | 4,11 | 0,49 |
| **Team Competence** | 3,87 | 0,88 |
| **Time and Time Constraint** | 3,55 | 0,85 |
| **Waterfall Model and Agile Approach in Software Projects** | 3,82 | 0,37 |

In order to determine whether the differences of the waterfall model and agile approach in software projects of the participants according to the total work experience groups are significant or not, the anova test was performed. According to this; The difference between productivity and quality perceptions in the software produced is significant according to the total work experience groups.

The difference between the perceptions of the waterfall model and agile approach in software projects was significant compared to the total work experience groups. It was determined that this hypothesis was accepted in terms of productivity and quality sub-dimensions and scales.

**Table 3:** The waterfall model and agile approach of the participants in software projects differ according to the total work experience groups.

| Sub-Dimensions | Total Work Experience | n | $\bar{X}$ | s.s | F | p | Scheffe |
|---|---|---|---|---|---|---|---|
| **Teamwork and Motivation** | 0-5 years | 34 | 3,75 | 0,51 | 0,687 | 0,561 | |
| | 6-10 years | 65 | 3,79 | 0,49 | | | |
| | 11-15 years | 34 | 3,86 | 0,57 | | | |
| | 16 years and above | 12 | 3,63 | 0,22 | | | |
| **Project Requirements** | 0-5 years | 34 | 3,79 | 0,99 | 2,675 | 0,050 | |
| | 6-10 years | 65 | 3,75 | 0,98 | | | |
| | 11-15 years | 34 | 3,60 | 1,08 | | | |

10.5824/ajite.2023.03.002.x

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 16 years and above | 12 | 4,52 | 0,53 | | | |
| **Efficiency and Quality** | 0-5 years(1) | 34 | 4,12 | 0,48 | 4,513 | 0,005* | (4-1) (4-2) (4-3) |
| | 6-10 years(2) | 65 | 4,13 | 0,50 | | | |
| | 11-15 years(3) | 34 | 3,94 | 0,49 | | | |
| | 16 years and above(4) | 12 | 4,53 | 0,26 | | | |
| **Team Competence** | 0-5 years | 34 | 3,90 | 0,81 | 2,119 | 0,101 | |
| | 6-10 years | 65 | 4,05 | 0,81 | | | |
| | 11-15 years | 34 | 3,59 | 0,97 | | | |
| | 16 years and above | 12 | 3,75 | 1,10 | | | |
| **Time and Time Constraint** | 0-5 years | 34 | 3,46 | 0,86 | 0,880 | 0,453 | |
| | 6-10 years | 65 | 3,60 | 0,94 | | | |
| | 11-15 years | 34 | 3,47 | 0,77 | | | |
| | 16 years and above | 12 | 3,88 | 0,53 | | | |
| **Waterfall Model and Agile Approach in Software Projects** | 0-5 years(1) | 34 | 3,82 | 0,36 | 2,790 | 0,043* | (3-2) (3-4) |
| | 6-10 years(2) | 65 | 3,85 | 0,39 | | | |
| | 11-15 years(3) | 34 | 3,69 | 0,39 | | | |
| | 16 years and above(4) | 12 | 4,03 | 0,19 | | | |

The waterfall model and agile approach of the participants in software projects differ according to company sector groups. According to the Anova test, the productivity and quality perceptions of those whose firm sector is **Production** are higher.

**Table 4:** The waterfall model and agile approach of the participants in software projects differ according to company sector groups.

| Sub-Dimensions | Company Sector | n | $\bar{X}$ | s.s | F | p | Scheffe |
|---|---|---|---|---|---|---|---|
| **Teamwork and Motivation** | Banking/Finance | 29 | 3,76 | 0,47 | 0,729 | 0,536 | |
| | Airline/Aviation | 56 | 3,74 | 0,53 | | | |
| | Production | 15 | 3,73 | 0,28 | | | |
| | Other | 45 | 3,88 | 0,53 | | | |
| **Project Requirements** | Banking/Finance | 29 | 4,00 | 0,96 | 1,856 | 0,140 | |
| | Airline/Aviation | 56 | 3,77 | 0,85 | | | |
| | Production | 15 | 4,13 | 0,94 | | | |
| | Other | 45 | 3,56 | 1,16 | | | |
| **Efficiency and Quality** | Banking/Finance(1) | 29 | 4,15 | 0,52 | 4,676 | 0,004* | (3-1) (3-2) (3-4) |
| | Airline/Aviation(2) | 56 | 4,04 | 0,50 | | | |
| | Production(3) | 15 | 4,53 | 0,48 | | | |
| | Other(4) | 45 | 4,04 | 0,41 | | | |
| **Team Competence** | Banking/Finance | 29 | 3,84 | 0,98 | 0,426 | 0,735 | |
| | Airline/Aviation | 56 | 3,92 | 0,85 | | | |
| | Production | 15 | 4,07 | 0,86 | | | |
| | Other | 45 | 3,79 | 0,90 | | | |
| **Time and Time Constraint** | Banking/Finance | 29 | 3,53 | 0,84 | 0,392 | 0,759 | |
| | Airline/Aviation | 56 | 3,48 | 0,94 | | | |
| | Production | 15 | 3,57 | 0,82 | | | |
| | Other | 45 | 3,67 | 0,78 | | | |
| **Waterfall Model and Agile Approach in Software Projects** | Banking/Finance | 29 | 3,84 | 0,44 | 1,051 | 0,372 | |
| | Airline/Aviation | 56 | 3,79 | 0,37 | | | |
| | Production | 15 | 3,97 | 0,34 | | | |
| | Other | 45 | 3,79 | 0,36 | | | |

It was determined that the differences between the waterfall model and agile approach in software projects of the participants were significant according to the average project duration groups. Those with an average project duration of **1-3 months** have a higher perception of **Time and time constraints** than the others. It was determined that this hypothesis was accepted in terms of Time and time constraint sub-dimensions.

**Table 5:** The waterfall model and agile approach of the participants in software projects differ according to the average project duration groups.

| Sub-Dimensions | Average Project Duration | n | $\bar{X}$ | s.s | F | p | Scheffe |
|---|---|---|---|---|---|---|---|
| **Teamwork and Motivation** | 1-3 months | 55 | 3,75 | 0,59 | 0,310 | 0,734 | |
| | 4-6 months | 45 | 3,83 | 0,44 | | | |
| | 7 months and above | 45 | 3,78 | 0,43 | | | |
| **Project Requirements** | 1-3 months | 55 | 3,78 | 0,87 | 0,275 | 0,760 | |
| | 4-6 months | 45 | 3,87 | 0,99 | | | |
| | 7 months and above | 45 | 3,72 | 1,15 | | | |
| **Efficiency and Quality** | 1-3 months | 55 | 4,12 | 0,51 | 0,002 | 0,998 | |
| | 4-6 months | 45 | 4,12 | 0,54 | | | |
| | 7 months and above | 45 | 4,11 | 0,43 | | | |
| **Team Competence** | 1-3 months | 55 | 3,99 | 0,76 | 1,563 | 0,213 | |
| | 4-6 months | 45 | 3,69 | 1,02 | | | |
| | 7 months and above | 45 | 3,93 | 0,88 | | | |
| **Time and Time Constraint** | 1-3 ay(1) | 55 | 3,83 | 0,75 | 5,368 | 0,006* | (1-2) |
| | 4-6 ay(2) | 45 | 3,29 | 0,88 | | | |
| | 7 months and above(3) | 45 | 3,50 | 0,87 | | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Waterfall Model and Agile Approach in Software Projects** | 1-3 months | 55 | 3,85 | 0,41 | 0,380 | 0,685 | |
| | 4-6 months | 45 | 3,79 | 0,37 | | | |
| | 7 months and above | 45 | 3,82 | 0,35 | | | |

According to the results of the Kruskal-Walis H test, which was conducted to test whether the difference between the customer satisfaction of the participants according to the variable of the company industry is significant, the difference of **Customer Satisfaction** according to the industry of the company is significant. According to the results of the Mann-Whitney U test, which was conducted to determine between which groups the perception differs in customer satisfaction, the customer satisfaction perceptions of those whose company sector is **Production** are higher.

**Table 6:** The customer satisfaction of the participants differs according to the company sector groups.

| **Sub-Dimensions** | **Sector** | **n** | **Mean Rank** | **$\chi 2$** | **p** | **Scheffe** |
|---|---|---|---|---|---|---|
| **Customer Satisfaction** | Banking/Finance | 29 | 55,36 | 15,254 | 0,002* | (1-4) |
| | Airline/Aviation | 56 | 79,54 | | | |
| | Production | 15 | 100,70 | | | |
| | Other | 45 | 67,00 | | | |

According to the independent T-test conducted to determine whether the differences between the waterfall model and agile approach in software projects of the participants are significant according to the number of people in the software team, the number of people in the software project teams showed a significant difference.

**Table 7:** The waterfall model and agile approach of the participants in software projects differ according to the number of people in the software project teams.

| **Sub-Dimensions** | **Number of People in Software Project Teams** | **n** | **$\bar{X}$** | **s.s** | **t** | **sd** | **p** |
|---|---|---|---|---|---|---|---|
| **Teamwork and Motivation** | 7 people and below | 62 | 3,77 | 0,51 | -0,356 | 143 | 0,723 |
| | over 7 people | 83 | 3,80 | 0,49 | | | |
| **Project Requirements** | 7 people and below | 62 | 3,77 | 0,92 | -0,161 | 143 | 0,872 |
| | over 7 people | 83 | 3,80 | 1,05 | | | |

10.5824/ajite.2023.03.002.x

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Efficiency and Quality** | 7 people and below | 62 | 4,12 | 0,58 | 0,067 | 143 | 0,947 |
| | over 7 people | 83 | 4,11 | 0,42 | | | |
| **Team Competence** | 7 people and below | 62 | 3,87 | 0,85 | -0,097 | 143 | 0,923 |
| | over 7 people | 83 | 3,89 | 0,92 | | | |
| **Time and Time Constraint** | 7 people and below | 62 | 3,57 | 0,89 | 0,169 | 143 | 0,866 |
| | over 7 people | 83 | 3,55 | 0,84 | | | |
| **Waterfall Model and Agile Approach in Software Projects** | 7 people and below | 62 | 3,80 | 0,39 | -0,491 | 143 | 0,624 |
| | over 7 people | 83 | 3,83 | 0,38 | | | |

Whether the waterfall model and agile approach of the participants in software projects differ according to the working position groups was tested with the Anova test. Accordingly, the waterfall model and agile approach in software projects of the participants do not show any difference according to the position groups studied.

**Table 8:** The waterfall model and agile approach of the participants in software projects differ according to the position groups worked.

| Sub-Dimensions | Position | n | $\bar{X}$ | s.s | F | p |
|---|---|---|---|---|---|---|
| **Teamwork and Motivation** | Business/Systems Analyst | 42 | 3,78 | 0,51 | 0,358 | 0,784 |
| | Software developing expert | 61 | 3,75 | 0,51 | | |
| | Manager | 17 | 3,82 | 0,48 | | |
| | Other | 25 | 3,86 | 0,48 | | |
| **Project Requirements** | Business/Systems Analyst | 42 | 3,89 | 1,02 | 0,226 | 0,879 |
| | Software developing expert | 61 | 3,75 | 1,06 | | |
| | Manager | 17 | 3,69 | 1,11 | | |
| | Other | 25 | 3,80 | 0,69 | | |
| **Efficiency and Quality** | Business/Systems Analyst | 42 | 4,25 | 0,41 | 1,597 | 0,193 |
| | Software developing expert | 61 | 4,03 | 0,57 | | |
| | Manager | 17 | 4,08 | 0,40 | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | Other | 25 | 4,12 | 0,46 | | |
| **Team Competence** | Business/Systems Analyst | 42 | 3,98 | 0,96 | 0,882 | 0,452 |
| | Software developing expert | 61 | 3,75 | 0,96 | | |
| | Manager | 17 | 4,09 | 0,64 | | |
| | Other | 25 | 3,88 | 0,70 | | |
| **Time and Time Constraint** | Business/Systems Analyst | 42 | 3,62 | 0,79 | 0,211 | 0,888 |
| | Software developing expert | 61 | 3,53 | 0,92 | | |
| | Manager | 17 | 3,44 | 1,06 | | |
| | Other | 25 | 3,60 | 0,65 | | |
| **Waterfall Model and Agile Approach in Software Projects** | Business/Systems Analyst | 42 | 3,88 | 0,37 | 0,736 | 0,532 |
| | Software developing expert | 61 | 3,77 | 0,42 | | |
| | Manager | 17 | 3,80 | 0,40 | | |
| | Other | 25 | 3,85 | 0,25 | | |

## CONCLUSION

Within the scope of the study, a comparison of the agile methods and the waterfall model was made and it was tried to determine how software developers perceived agile approaches. In order to compare the result obtained with the experimental experience, a questionnaire was applied with the participation of 145 employees from the sectors who took part in agile projects.

The survey consists of two main parts. The first part was prepared to collect demographic information about the participants. In the second part, questions were asked about the perception of agile software development. According to the survey results, SCRUM comes to the fore as the commonly used approach. The results significantly overlap with the results of the study carried out by Agile Turkey in 2019. (Agile Turkey,2019)

The attitudes of the software developers participating in the study on agile approaches are quite positive. They think that agile approaches increase efficiency, quality, and customer satisfaction. Participants with total experience and industry experience of 16 years or more have higher perceptions of Agile approach productivity and quality. In addition, according to the survey results, those who think that the Agile approach is suitable for small teams are in the majority.

Since the waterfall model is carried out depending on the past phases of the project, it is retrospective in understanding and is an order in which all project elements are expected to do the work defined for them. The waterfall model, which makes use of this layout, is more

10.5824/ajite.2023.03.002.x

suitable for projects where the requirements and goals are clear, the solutions and the work to be done are known, the requirements will not change, the software to be produced has no fault tolerance and has functional integrity. Defense systems and embedded systems can be given as the best usage area.

Agile methods, on the other hand, have a forward-looking understanding due to their structure that encourages change and feedback. Autonomous teams are expected to do everything by themselves and benefit from cross-functionality and experimentation in this process. This requires an approach to creativity rather than order. Therefore, agile methods are more suitable for projects where requirements and goals are uncertain and uncertain, change is frequent and value generation is desired in the short term. The best use for agile methods is web-based workflow management systems. In this direction, according to the results of the survey, the productivity and quality perceptions of those whose firm sector is the production of agile methods are higher than those who are in the banking/finance, airline/aviation, and other sectors.

The survey result evaluations were made according to the statistical findings determined in order to maintain impartiality; However, as in many experimental studies, there are important limitations in this study. First of all, the participants who answered the questionnaire could not be selected by random sampling across Turkey.

Thus, the results cannot be generalized to the whole situation in Turkey, but they can provide software developers and researchers with a perspective on general trends in Turkey. Within the scope of future studies, dissemination of the research over a longer period of time to cover a larger sample or re-application of the survey at periodic intervals will be beneficial in terms of expanding the findings.

## REFERENCES

Agile Turkey. (2019). 9th Annual Agility Report Retrieved from: https://www.agileturkey.org/reportopt.pdf

Ahmad, O. M., Markkula, J. & Ovio, M., (2013). Kanban in software development: *A systematic literature review*. Finland, 39th Euromicro Conference Series on Software Engineering and Advanced Applications. https://doi.org/10.1109/seaa.2013.28

Ahmed, A. (2012). *Software Project Management A Process-Driven Approach*, Florida: CRC Press

Awad, M. A. (2005). *A Comparison between Agile and Traditional Software Development Methodologies*, Published Dissertation: the University of Western Australia.

Balaji, S., Murugaiyan, M. S. (2012), Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. I*nternational Journal of Information Technology and Business Management.* Retrieved from: www.jitbm.com

Butler Jr. A. G. (1973). Project Management: A Study in Organizational Conflict. *The Academy of Management Journal*, 16(1), 84-101. https://doi.org/10.5465/255045

Cervone, H. F. (2011), Understanding agile project management methods using Scrum. *OCLC Systems & Services: International Digital Library Perspectives*, 27(1), 18-22. https://doi.org/10.1108/10650751111106528

Cleland, D. I., Gareis R. (2006), *Global Project Management Handbook: Planning, Organizing, and Controlling International Project,* (Twelfth Edition), the United States of America: The McGraw-Hill Companies

Cockburn A.(2000-2001). *Agile Software Development* (Draft version:3b). Highsmith Series Editors.

Farrell, A. (2007). Selecting a Software Development Methodology based on Organizational Characteristics: *An Essay Submitted in Partial Fulfillment of the Requirements for the Degree of "Master of Science in Information Systems",* Athabasca University, Alberta

Flora, H. K., Chande, S. V. (2014). A Systematic Study on Agile Software Development Methodologies and Practices. *International Journal of Computer Science and Information Technologies*, 5(3), ISSN: 0975-9646

Gechman M. (2019). *Project Management of Large Software-Intensive Systems Controlling the Software Development Process*, Florida: CRC Press

Heldman K. (2018). *PMP Project Management Professional Exam Study Guide,* Indiana: Sybex

Hewlett-Packard Enterprise. (2017). Agile is the new normal: *Adopting Agile project management.* Retrieved from: https://softwaretestinggenius.com/docs/4aa5-7619.pdf

Highsmith, J. (2000). Adaptive Software Development: A Collaborative Approach to Managing Complex Systems, Dorset House.

Horine, G.(2009). *Absolute Beginner's Guide to Project Management*, (Second Edition), Indiana: Que Publishing

IEEE Computer Society. (2014). *SWEBOK: Guide to the Software Engineering Body of Knowledge,* USA: IEEE

Kerzner, H. (2017). *Project Management a Systems Approach to Planning, Scheduling, and Controlling*, (Twelfth Edition), Hoboken, New Jersey: Wiley

Lock, D. (2007). *Project Management*, (Ninth Edition), Hampshire: Gower Publishing

Moran, A. (2015). *Managing Agile. Strategy, Implementation, Organisation, and People*, Zurich: Springer

N.M.A., Govardhan, A. (September 01, 2010). A Comparison Between Five Models Of Software Engineering. *International Journal of Computer Science Issues*, 7, 5, 94-101.Packard Enterprise, U.S.A.

Özdemir, E., (2017). Waterfall Yazılım Geliştirme. Retrieved from: http://erenozdemir.net/waterfall-yazilim-gelistirme/

Palmquist, M. S., Lapham, M. A., Miller, S., Chick, T., & Ozkaya, I. (2013). Parallel worlds: Agile and waterfall differences and similarities. *Carnegie-Mellon University Pittsburgh Pa Software Engineerıng Inst.* (No. CMU/SEI-2013-TN-021).

Poppendieck M.,Poppendieck T. (2013). *The Lean Mindset Ask the Right Questions,* Massachusetts: Addison-Wesley Professional

Pressman R. S., Maxim B. R. (2019). *Software Engineering:A Practitioner's Approach* (9th Edition), New York: Mc Graw Hill

Project Management Institute (PMI). (2000), *A Guide To The Project Management Body Of Knowledge* (PMBOK GUIDE)(2000 edition), Pennsylvania: PMI

Project Management Institute (PMI). (2013), *Software Extens,on to the PMBOK Guide* (6th Edition), Pennsylvania: PMI

Project Management Institute (PMI). (2017), *A Guide To The Project Management Body Of Knowledge* (PMBOK GUIDE)(6th Edition), Chicago: PMI

Project Management Institute (PMI). (2017). Agile Practice Guide (PMBOK GUIDE), (First Edition), Pennsylvania: PMI

Rajagopalan , S.,Mathew, S. K. (2016). Choice of Agile Methodologies in Software Development. *International Information Management Association*, 25 (ISSN:1941-6679-Online Copy)

Ruhe, G.,Wohlin, C. (2014). *Software Project Management in a Changing World*, Berlin:Springer

Sergeev, A., (2016), Team Roles in Waterfall Methodology. Retrieved from: https://hygger.io/blog/team-roles-in-waterfall-methodology/

Schwaber K., Sutherland J. (2017), The Scrum Guide, The Definitive Guide to Scrum: The Rules of the Game. Retrieved from: www.scrum.org

Schwaber K., Sutherland J. (2020), The Scrum Guide, The Definitive Guide to Scrum: The Rules of the Game. Retrieved from: www.scrum.org

Slack, N., Chambers, S., Johnston, R. (2007). *Operations Management*, (Fifth Edition), London: Financial Times/Prentice Times

Sommerville I. (2007). *Software Engineering 8*, Harlow: Pearson Education Limited, Eighth edition

Stanley, J. C., Gross, E. D. (2020). *Project Management Handbook: An Introduction to Scrum, Agile and DevOps for Absolute Beginners*, Prosper Consulting Inc.

Stepanek, G. (2005). *Software Project Secrets: Why Software Projects Fail*, Berkeley: Apress.

VersionOne Inc. (2020). 14th Annual State of Agile Report, U.S.A.:Analysis.Net Research. Retrieved from: https://digital.ai/resource-center/analyst-reports/state-of-agile-report

Villafiorita A. (2014). *Introduction to Software Project Management*, Florida: CRC Press

Vohra, P., Singh, A. (2013). A Contrast and Comparison of Modern Software Process Models. *International Conference on Advances in Management and Technology*. (Icamt 2013). 23-27.

Weiss, J. W. , Wysocki, R. K.(1992). *5-Phase Project Management A Practical Planning&Implementation Guide*, The United States of America: Perseus Books Publishing