MASTER‘S THESIS

Agile Project Management: MSD Case Study

STUDY DEGREE PROGRAMME

Innovation Project Management

THESIS SUPERVISOR

Ing. Mgr. Tomáš Sadílek, Ph.D.

NICOLE

VOSÁTKOVÁ

2023

MASTER‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Vosátková Nicole** Personal ID number: **475202**

Faculty / Institute: **Masaryk Institute of Advanced Studies**

Department / Institute: **Institute of Management Studies**

Study program: **Innovation Project Management**

II. Master’s thesis details

Master’s thesis title in English:

Agile Project Management: MSD Case Study

Master’s thesis title in Czech:

Agilní projektové řízení: případová studie společnosti MSD

Guidelines:

AIM: What agile practices are commonly used, and what are the most effective?
CONTRIBUTIONS: What are the main benefits of an agile approach? What are the significant barriers to adopting agile practices? What are the critical factors of successfully implementing an agile approach? What impact are the agile development practices having on work outcomes?
CONTENT: 1. Introduction; 2. Theoretical part - Project Management Development Methods, Agile; 3. Practical part - Introduction of the Company, MSD, Research Methodology, Analysis of Results, Evaluation and Recommendations; 4. Conclusion
METHODS: Selected Project Management Methods (Lean, Waterfall), and Agile Methods (Scrum, Kanban), Agile Terminology: Agile Roles (Agile Team, Scrum Master, Product Owner, Owner Proxy), Agile Tools (Mural, Jira, MS Software Applications), Agile Events (Sprints, Meetings).

Bibliography / sources:

CLUSTER, Konnor. Agile Project Management: Learn How to Manage a Project with Agile Methods, Scrum, Kanban and Extreme Programming. Independently Published, [2019]. ISBN 9781710291698.
COLE, Rob. Agile Now: Your quick start introduction to agile. Pearson Business, [2021]. ISBN 9781292232003.
MORAN, Alan. Managing Agile - Strategy, Implementation, Organisation and People. Springer, [2015]. ISBN 978-3-319-16261-4.

Name and workplace of master’s thesis supervisor:

Ing. Mgr. Tomáš Sadílek, Ph.D. Masarykův ústav vyšších studií ČVUT v Praze

Name and workplace of second master’s thesis supervisor or consultant:

Date of master’s thesis assignment: **09.12.2022** Deadline for master's thesis submission: **27.04.2023**

Assignment valid until: _____

Ing. Mgr. Tomáš Sadílek, Ph.D. Supervisor’s signature	Ing. Dagmar Skokanová, Ph.D. Head of department’s signature	prof. PhDr. Vladimíra Dvořáková, CSc. Dean’s signature

III. Assignment receipt

The student acknowledges that the master’s thesis is an individual work. The student must produce her thesis without the assistance of others, with the exception of provided consultations. Within the master’s thesis, the author must state the names of consultants and include a list of references.

. _____
Date of assignment receipt

_____
Student’s signature

VOSÁTKOVÁ, Nicole. *Agile Project Management: MSD Case Study*. Praha: CTU 2023. Master‘s thesis. Czech Technical University in Prague, Masaryk Institute of Advanced Studies.

MASARYK
INSTITUTE
OF ADVANCED
STUDIES
CTU IN PRAGUE

Statement

I declare I have accomplished my final thesis by myself and I have named all the sources used in accordance with the Guideline on ethical preparation of university final theses.
I have no relevant reason against using this schoolwork in the sense of Act No. 121/2000 Coll. on Copyright and Rights Related to Copyright and on Amendment to Certain Acts (the Copyright Act).

In Prague: April 25, 2023 Signature:

Acknowledgement

I would like to express my sincere gratitude to all those who have helped me in the development of my thesis. I am especially grateful to Ing. Mgr. Tomáš Sadílek, Ph.D., thesis supervisor, for his invaluable guidance, unwavering support, and constructive feedback throughout the process. His expertise and dedication have been instrumental in shaping the final outcome of this thesis.

I would also like to extend my appreciation to MSD, especially my manager Ing. Tomáš Kejzlar, and the entire Agile team for their insightful advice, encouragement, and assistance in writing this thesis. Their contributions have greatly enriched my understanding of the subject matter and helped me to approach the research with greater depth and clarity.

Additionally, I would like to acknowledge the valuable contributions made by Lin Fan (François Bouchetoux) during my study stay in Taiwan. His collaboration and feedback have been immensely beneficial in shaping the direction of this thesis.

Finally, I would like to express my heartfelt gratitude to all those who have offered their support and assistance in any way during the writing of this thesis. Your encouragement and motivation have been greatly appreciated and have helped me to overcome the challenges that arose along the way.

Abstract

This thesis examines the effectiveness of Agile Project Management, an interactive project management method, from the perspective of employees at MSD, a leading global biopharmaceutical company. The aim of this study is to analyse and describe the use of Agile methodology by MSD's IT staff. The theoretical part of the thesis compares traditional project management with Agile methodology. The practical part of the thesis is based on a qualitative method, a Case Study, including a quantitative survey, to analyse the use of Agile methodology among MSD IT employees and to understand their views and experiences of implementing Agile management. The findings indicate MSD IT staff frequently employ various Agile practices, particularly product backlog, the presence of a Product Owner, as well as the utilisation of either a Kanban or Scrum board. Moreover, the Agile practices deemed most effective by MSD IT staff entail the presence of a Scrum Master, a Product Owner, daily collaboration, as well as a product backlog. Based on the findings, recommendations are made for MSD to enhance their Agile methodology and increase awareness of Agile within MSD IT. These recommendations encompass offering broader educational programs, training, and support to employees regarding Agile practices and enhancing awareness and promoting the availability of such resources. In addition, cultivating a culture that prioritises ongoing enhancements and fostering cross-functional cooperation. Overall, this thesis contributes to the understanding of Agile methodology and its effectiveness in the context of MSD IT. Additionally, the conclusions and recommendations presented in this study may also be useful for other companies seeking to implement Agile methodology in their organisations.

Key words

Project Management, Project Management Methodologies, Agile, Agile Project Management, Agile Approach, Agile Development Methods, Agile Frameworks, Agile Coaching, MSD Case Study

Content

Abbreviations and Symbols

- **ICT** = Information and Communication Technology
- **LeSS** = Large-Scale Scrum
- **OKRs** = Objectives and Key Results
- **PMI** = Project Management Institute
- **PMP** = Project Management Professional
- **SAFe** = Scaled Agile Framework
- **SOW** = Statement of Work
- **R&D** = Research and Development
- **USD** = United States Dollar
- **WBS** = Work Breakdown Structure
- **WIP** = Work in Progress
- **XP** = Extreme Programming

Introduction

In the world we live in today, our surroundings are always changing. In this age of globalisation and digital technology, businesses are under increasing pressure to complete projects as quickly and affordably as possible and to be able to adapt to the environment's changing needs. As a result, firms are increasingly attempting to incorporate aspects of Agile project management rather than traditional project management into their initiatives. This tendency is anticipated to continue in the future, along with the rising demands for project management, whether traditional or largely Agile approaches. This is due to the ever-increasing rate of project management in nations throughout the world.

With the development of communication services and the introduction and gradual spread of available information technologies in the mid-1990s, the design and development of products and services underwent a fundamental transformation. Programs and products that had previously been planned out over a long period of time were suddenly confronted with very quick technological advances for which they were unprepared and were frequently outmoded by the time they were released. In a global setting with quick information sharing, the direction of development was not always foreseeable, and the market's response abruptly deviated from historical precedent and established patterns.

Project management has had to adapt to this trend. Unconventional methods have been used, particularly in rapidly evolving industries like IT, software engineering, the automotive sector, and others, from which specific, more flexible project management principles have developed over time. These new, so-called Agile management techniques have the flexibility to respond to changes that arise during the project itself and to modify one or more parameters in accordance with actual requirements. However, even though Agile methods are most frequently utilised and known in the IT industry, where they help manage individual projects or entire companies; nowadays, this approach is also applied in many other industries, where both small companies and multinational corporations, as in a biopharmaceutical company MSD, want to use Agile to achieve a higher level of quality and competitiveness in the current market.

The aim of this thesis is to clarify and explain the Agile theory, including a case study that explores the implementation of Agile project management in a multinational company MSD, where the author of this thesis is currently working as an intern for the Agile coaching team, which is one of the reasons for choosing this topic. Furthermore, although Agile has been popular for a longer time and is becoming more and more popular in the business world, not everybody has heard of Agile. The main target of the case study is to focus on the employees' perceptions of Agile management changes and the effectiveness of Agile at MSD. The case study examines the most used Agile methods/frameworks, practices, and Agile tools in MSD IT. Next, it also examines what Agile practices and benefits the employees consider as most important and focuses on their perceptions of the Agile adoption process – what barriers employees most often encounter during Agile adoption and, conversely, what are the important factors of this implementation.

This diploma thesis is useful primarily for the MSD organisation, which can benefit from the results of its first research and case study regarding the implementation of an Agile approach from the perspective of MSD IT employees. From the study, the company will find out how its employees perceive the introduction of the Agile approach in the company, what effects Agile has on their work outcomes, what to pay more attention to and what can help to manage the implementation without major complications. Nevertheless, the thesis can also be beneficial for other companies which are planning to implement Agile management in their projects and see from the results what Agile can bring for another company; but equally, it can be beneficial for companies that have already implemented Agile management in their projects and are looking for areas for improvement or for extending Agile management to other parts of the company. At the same time, the work can be generally beneficial for all people who have not heard much about Agile yet, so they would like to get knowledge about Agile theory and understand the difference and compare it with traditional approaches.

Despite the popularity of Agile project management and the fact that it is much talked about, many companies still use traditional management methods. MSD has been implementing Agile management effectively for a couple of years. In 2018–2019, there was a substantial surge in the adoption of Agile methodologies within the software development industry. This trend was exemplified by the creation of an internal software development team (Global Software Development, GSW), which served as a testament to the growing popularity of Agile practices. Subsequently, the organisation established its first official independent team of Scrum Masters and coaches, who were tasked with facilitating the adoption and implementation of Agile methodologies across various departments. Moreover, another team of product managers was created, further highlighting the organisation's commitment to embracing Agile principles and practices. Nowadays, MSD is engaged in Agile development, has several teams of professional Agile coaches, Scrum Masters and product managers, and organises various training sessions for its employees. Nevertheless, the question is how MSD employees perceive the implementation of an Agile approach; “What Agile practices are commonly used by MSD staff, and what are the most effective?". This thesis aims to answer these main research questions and other research questions, describe the traditional project management and well-known methods, explain theory about Agile, how Agile project management is done and mention popular Agile techniques, frameworks, and tools. In conclusion, the purpose of this thesis is to explain the Agile theory and principles from the literature perspective and then conduct the MSD case study to answer the given research questions.

The thesis is divided into two parts, theoretical and practical. The theoretical part will introduce project management, the project and traditional project management methods. Then, it is focused on describing Agile, Agile team roles and tools, and Agile project management in comparison to traditional project management. Further are theoretically described Agile methods and techniques, transformation to Agile methods, where common reasons for implementing Agile and its benefits as well as challenges and barriers to Agile adoption are mentioned. The thesis also focuses on a detailed description of the two most commonly used Agile methods, which are Scrum and Kanban. In the practical part, firstly, the company MSD will be introduced, then, the research

methodology including the research questions of the MSD case study are formulated. Simultaneously, the research strategy and design will be described, the core and sample research population will be identified, and the choice of methods used in the practical part of the thesis will be discussed. Subsequently, the data collected will be analysed. At the end, an overview will be given where the data will be summarised and based on the data collected, proposals and recommendations for the MSD organisation will be determined.

Overall, the whole practical part focuses on the qualitative method, a case study about the MSD company, where a quantitative questionnaire method was used for the research of Agile effectiveness and outcomes in the biopharmaceutical corporate company MSD. Subsequently, the results obtained from the questionnaire will be interpreted and briefly compared with the other public studies such as the latest annual State of Agile Report, 2022, published by Digital.ai. Finally, the research questions will be answered. At the end, the practical part aims to provide proposals and recommendations based on the data analysis of the case study and to evaluate the benefits of the proposed solution.

CHARACTERISTICS OF PROJECT MANAGEMENT AND AGILE

- THEORETICAL PART

1 Project Management

Effective project management has become increasingly important in today's business environment. This chapter provides an introduction to the field of project management, including its purpose, benefits, and underlying concepts. We discuss critical components of project management, including project definition, the project management triangle, the project life cycle, and the role of the project manager. Project definition is essential to provide a clear understanding of the project's goals and objectives. The project management triangle defines the relationship between time, cost, and scope in project management. The project life cycle identifies the key phases of a project. The project manager is responsible for managing project resources and ensuring that the project is completed on time and within budget. This chapter aims to provide readers with a comprehensive understanding of project management principles and practices. By the end of this chapter, readers will have a firm understanding of project management, giving them a greater understanding of what project management actually entails.

1.1 Introduction to Project Management

Project management is unavoidable in today's world, where there is constant progress through numerous types of projects. In addition to being necessary for improvement, project management is a field that actively works to advance itself by influencing several PM success variables (Radujkovic, Sjekavica, 2017).

In the discipline of project management, methods for managing projects are established and improved based on predefined parameters. American engineer and management consultant prof. Harold Kerzner, defines project management as a collection of actions that involves planning, organising, directing, and controlling the resources of a company that have been formed with a relatively short-term purpose to fulfil specified goals and objectives (Kerzner, 2017, p. 2).

The definition of project management, as stated by the Project Management Institute (PMI), involves the utilisation of an extensive range of abilities, methodologies, and instruments to accomplish the necessary objectives of a particular project (PMI, 2017, p. 5).

Project management is used to plan and execute complex, usually one-off actions that need to be carried out within the required timeframe at the planned cost to achieve the set objectives. Briefly, project management can also be characterised as the efficient and effective delivery of change. ***The subject of project management*** is the ***project***, which is a set of activities that need to be planned and executed to achieve the desired objectives, outputs. ***The aim of project management*** is to ensure the planning and implementation of a successful project, which might be defined as a case where the objectives of the project have been achieved in the planned time and at the planned cost. Change is caused by the implementation of the project output. Usually, we cannot implement the change directly, but we assume that the implementation of the project will cause the implementation of the change. Project management is based on the understanding that

once the scope, unusualness, complexity, difficulty, and risk of a project exceeds a certain level, adequate methods need be used to manage the entire event. In addition, we can identify two other principles that project management uses, namely the ***principle of teamwork***, where the joint work of different specialists can solve even very complex problems, and the ***principle of systematic work***, which is based on exact methods (Doležal, Máchal, Lacko, 2012, p. 21–22).

Project management differs from the usual form of operational management in a line-managed company mainly in its temporariness and in the allocation of resources for its implementation according to the needs of the project. If the project objectives are achieved, the project ends; if the objectives are achieved in operational management, new objectives are set, and work continues. Labour, financial, or technological resources are planned and allocated to the project according to its planned needs, and these resources are consumed or transferred to other projects when the project ends. In general, project management is a tool for implementing defined changes that are very difficult to implement other than through a project, which is a set of activities that form a path from the initial state to the defined desired target, target state. To ensure that all project management practices are applied is the responsibility of the project manager. Successful project management means that the planned ***objective*** of the project is achieved in the planned ***time***, using, and adhering to the projected ***costs*** in a certain degree of uncertainty, risk, and quality of implemented inputs (Svozilová, 2011, p. 20).

Besides, project management is based on the principles, methods, and tools of general management methods and uses knowledge from other disciplines such as psychology, economics, mathematics, and information technology. A key issue in contemporary management is managing in a context of constant change. Organisations of any orientation, and in any sector, must constantly respond to new stimuli, and adapt to changes or dynamic environments. The need to respond to new stimuli is nowadays becoming a key factor on the road to success. Influences for change in an organisation come from both the external and internal environment. From the external environment, the factors in particular that affect the organisation are ***political factors*** that influence organisations primarily through a change in legislation and business rules. Next, ***economic factors*** of greatest importance are changes in the growth (stagnation, decline) of gross domestic product, the rate of inflation, the tax burden, wage and social measures, limited resources for public administration. Then, ***social factors***, especially changes in population preferences, lifestyle, and consumer behaviour. And, ***science and technology development***, that not only develops information technology but also causes the emergence of new industries and the demise of some existing ones. Necessary innovations increase efficiency and produce more outputs for the same cost (Bartoš, Bartošová & Ponikelský, 2012, p. 10).

One of the important trends is the changes in approaches to management and organisational structures of companies, where high demands are placed on management, achieving goals limited by time barriers and budget or resource capabilities. Companies operate in different business environments, domestic, international, global, and competition is putting increasing pressure on management, managers, to improve production while reducing costs and transferring product or service placements to markets more quickly to meet customer needs. The need to respond to

turbulent developments and technological change is the cause of modified standardised and Agile approaches to management. Since the second half of the twentieth century, there has been a growing importance in the application of project-based management elements, developing new management techniques, and procedures in order to find an effective approach to managing high levels of change – a shift towards project-based management (Doležal, Máchal, Lacko, 2012, p. 21–22).

Therefore, organisations, companies, businesses, and institutions have to adapt very agilely to ever-changing conditions if they want to stay in business. In today's information-based society, this need is stronger and standard forms of management are lagging behind. That is why the field of project management has started to develop as a tool to implement the necessary changes that businesses and organisations have gradually started to implement. This involved not only various organisational changes, technology replacements, implementation of more efficient processes, etc., but also the development of new innovative products and the achievement of ambitious goals (Doležal, Máchal, Lacko, 2012, p. 22).

1.2 Project Definition

The term project can be encountered in almost any field nowadays. PMI states that a project is a temporary endeavour started with the intention of producing a special product, service, or outcome (PMI, 2017, p.3). The temporary nature of projects suggests that a project has a definite beginning and end. This characteristic is important because a large part of the project effort is devoted to meeting a predetermined schedule, whether the projects last minutes, hours, months, or years., The term “temporary” does not automatically indicate that the period of the project is short. It refers to the engagement of a project and its endurance. Temporary does not even commonly refer to the product, service, or outcome produced by the project; most projects are intended to produce a permanent effect. Otherwise, projects are carried out at all organisational levels. One or more persons, one organisational unit, or several organisational units from other organisations might all be a part of a project (Watt, 2014, p. 10, PMI, 2017, p.3).

The main goal of a project is to provide a service or product that did not exist before, which makes each project unique. Each project produces a certain good, service, or outcome, which may be a physical product or immaterial. A project may produce a ***product*** that can be either a part of another thing, an improvement to another product, or a finished item in itself; a ***service*** or the ability to provide a service (e.g., a business function that promotes production or distribution); a ***development*** in the existing product or service lines (e.g., a Six Sigma project undertaken to decrease faults); or a ***result***, such as a statement or record (e.g., a research project that produces knowledge that may be used to determine if a trend exists, or a new procedure will be beneficial to society). Examples of projects include, without being limited to, evolving a new item, service, or result; effecting a change in an organisation’s structure, procedures, style, or staffing; developing obtaining, or modifying an information system (hardware or software); conducting research with results that will be accurately recorded; constructing a building, industrial facility, or infrastructure;

or implementing, enhancing, or improving current business processes and procedures (Watt, 2014, p. 10, PMI, 2017, p.3–4).

Projects are constantly carried out by organisations to enhance performance. Senior executives refer to their projects' accomplishments generally as "project success" to both subjectively describe and objectively assess them. When discussing projects, project success and project management success are the two key success ideas. These two project success factors have certain things in common and others in common. The primary distinction is that project management success is determined by conventional measurements of time, cost, and quality performance, whereas project success is determined by the evaluation of the entire project goals achieved. However, because there are so many alternative models for project success and project management success, it is difficult to distinguish between them, largely due to their interdependence (Meredith, Zwikael, 2019, Radujkovic, Sjekavica, 2017).

The term success is perhaps the most beloved word among project managers. However, there are other ways to evaluate success. The success of a project can evolve over time. A project can be assessed over a longer period of time as conditions may change. As an example, the Los Angeles red line metro project is a good illustration. The red line metro project was successful in that it was completed quickly and according to plan, opening to the public eight months earlier than expected and staying under budget. Due to the fact that it only managed to draw 60,000 passengers in its first year as opposed to the anticipated one million, it was seen as a failure in terms of long-term usefulness. The outputs, such as tracks and stations, were delivered ahead of schedule and within budget, which helped the construction project management fulfil their objectives. Nonetheless, because it did not ease traffic congestion, which was the main project purpose, the project has been deemed a failure, and the following phases were abandoned (Meredith, Zwikael, 2019, Radujkovic, Sjekavica, 2017).

Both successful projects and projects with unsuccessful project management are feasible. In other words, if a project has achieved higher and longer-term goals, it might still be successful despite unsuccessful project management. When project management ends, short-term goals may not be achieved, but long-term outcomes may succeed because a broader range of goals, as opposed to the more limited set that project management entails, are met. The project's end occurs when its goals have been met, when it is abandoned because its goals will not or cannot be realised, or when there is no longer a need for it. The client, which might be a customer, sponsor, or champion, may also decide to end a project if they so choose (PMI, 2017, p.3–4, Radujkovic, Sjekavica, 2017).

Managing a project also commonly includes, without limits, recognising requirements; taking into account the various needs, worries, and expectations of the stakeholders in planning and executing the project; establishing, sustaining, and carrying out active, efficient, and cooperative communications among stakeholders; managing stakeholders to generate deliverables that match project requirements and strike a balance between conflicting project constraints such as scope, quality, time, budget, resources, and risks. Regarding constraints, the project management team

needs to pay attention to, can vary based on the particular attributes and conditions of the project. These characteristics are correlated, meaning that if one changes, at least one other is likely to be impacted as well. For instance, when the timetable is compressed, it is frequently necessary to increase the budget to add more resources and finish the same amount of work faster. In order to complete the project in less time and for the same money, the scope or the intended quality may be reduced if an increase in the budget is not an option. An even bigger problem could arise from project stakeholders' different perspectives on what variables are most crucial. Moreover, risks could increase if the project requirements or objectives are changed. The project team must be capable of assessing the situation, balancing the needs, and maintaining proactive contact with stakeholders to generate a successful project. Due to the possibility of change, creating the project management plan is an iterative process that is gradually developed throughout the course of the project's life cycle. Progressive elaboration is the process of progressively refining and enhancing a strategy as more precise data, specific details, and estimations become available. By using progressive elaboration, project management team can define work and manage it in more detail as the project progresses (Watt, 2014, p. 12–13, PMI, 2017, p. 5–6).

As mentioned before, projects are anticipated to be finished in the short-term on schedule, under budget, and in accordance with their intended scope. Martin Barnes referred to this group of three operational goals as the "iron triangle" in 1969. Success in project management, which took its place, is described as how well the project plan was carried out, particularly in terms of meeting its time, cost, and scope objectives. The iron triangle model, which will be discussed in the next subsection, was the original project management success model, and it has now been shown to account for a relatively small portion of overall project success (Meredith, Zwikael, 2019).

1.2.1 Project Management Triangle

Proper project management requires proper identification of requirements and detailed writing of all objectives expected from the project. If everyone understands the main purpose and benefits at the outset, it is much easier to stay on track to meet the given expectations, and at the same time potential conflicts can be better avoided. Nevertheless, despite careful preparation, every project manager struggles with project constraints that include cost, scope, quality, risk, resources, and time. Most often, however, only time, cost and scope are discussed. These factors are also called the triple constraint, which is one of the basic concepts of project management (Watt, 2014, p. 13).

A project management model is a triangle, commonly known as the iron triangle, which suggests that controlling the three constraints of cost, scope, and time results in a final output of high quality. These constraints are interconnected and need to continually be balanced. When one constraint is changed, it may be necessary to modify one or both other constraints to preserve quality. This model is also named the iron triangle, golden triangle, project triangle or triple constraint (Rudder, Forbes Media LLC, 2022, Watt, 2014, p.13–14).

Project management triangle might be also defined as a project tri-imperative which is a graphical model of the basic project parameters that are plotted as distances on three axes in the plane. By joining these three points, a triangle is formed that expresses the interdependence of the individual variables. Hence, a change in one of the parameters leads to a change in at least one of the others (Doležal et al., 2012, p. 66).

Project managers employ a project management triangle to help their teams produce high-calibre final deliverables. The triangle provides a clear understanding of the correlation between scope, cost and time as it related to quality. By handling the three constraints, project managers, their teams and their clients gain many benefits, such as fewer project risks, simpler change management and improved client communication. In the end, the project is more likely to provide a high-quality final output if these factors are well managed (Rudder, Forbes Media LLC, 2022).

Hence, project managers are better able to manage risks and change easier. For example, if a project has a strict deadline and a change in scope occurs, project managers are aware that they should look at the third constraint to relieve the pressure. They can accomplish this by, for instance, adding talent to speed the project along in spite of a scope expansion. As such, the triangle provides special insights on how to manage change effectively and efficiently. Thanks to the triangle, project managers are also better in effective identification and managing project priorities. Knowing the three constraints in a project management triangle makes it simpler to identify and manage priorities to produce a high-quality final delivery. Consider a project with a limited budget. The project manager is aware of the need to closely monitor the project's scope and schedule given the constraints. After all, if the scope expands, the budget will likely have to raise as well. Additionally, added time calls for more paid resources, such salaries, to cover that time. Moreover, one of the most important things in project management is to communicate with clients easily, and the project management triangle helps to provide clients with clear communication. Clients depend on project managers to manage projects to quality outcomes. The knowledge of best practices in project management is expected. Consequently, they might not be very knowledgeable on the subject. Therefore, it is the project manager's responsibility to explain the effects of modifications to them in simple terms (Rudder, Forbes Media LLC, 2022).

The project triangle provides a simple illustration to help explain dependencies between cost, time, and scope as well as show how one change affects these constraints. These three mentioned advantages come together to reduce projects risk. Knowing which priorities are essential for efficient project management allows risk management planners to design plans around the priorities. To prevent one change from compromising the project's quality, change management plans can also consider how the three restrictions are related to one another. Finally, effective customer communication keeps client satisfaction high even when risks occur, preventing risks from getting worse (Rudder, Forbes Media LLC, 2022).

Figure 1 Project Management Triangle

Source: own processing according to: (Rudder, Forbes Media LLC, 2022)

The picture illustrated above *(Figure 1)* shows the project management triangle including three mentioned components – or constraints – that are often shown as the three sides to the triangle. As was previously mentioned, these include the project's scope, cost and time and they provide quality when balanced correctly. Hence, if one is out of balance, one side of the triangle may be too long or too short and the triangle may be broken. The triangle's ***scope*** component refers to the magnitude of the project being finished. The size is calculated based on the project's capacity, quality, complexity, quantity, or magnitude of deliverables, level of detail, and number of features (such as the number of users a software can support). The ***cost*** of a project includes any paid resources required for the project. Paid resources include human talent, raw materials, tools, facilities, inventory, and even important opportunities. Cost and time are typically closely related. The triangle's ***time*** component covers the project's overall timeframe, the number of hours each team member puts in, the intervals it takes to finish each milestone, and the time spent planning and strategizing. When managing time, it's common to need to change scope and budget (Rudder, Forbes Media LLC, 2022).

The ways to effectively manage a project management triangle will lean on the type of the project, its priorities, its propensity for risk and the experience and resources of a team. However, by a global platform for unbiased money advice, news and reviews, Forbes Advisor (Rudder, *Forbes Media LLC*, 2022), there are five strategies to consider when managing a project triangle and that is choosing a flexible constraint, listing features in order of most to least importance, making plans for managing changes and risks and matching a management methodology to the priority constraints. Although there are numerous project management methodologies to take into account, however some are more frequently utilised than others and one of the is Agile. Namely, three project management approaches that have been extensively examined and attempted are Waterfall, Agile, and Lean, which will be explained and discussed in later chapters.

1.2.2 Project Life Cycle

According to Svozilová (2011, p. 37), a project is an element that is procedural in nature, at the time of its existence, it evolves and is in different phases, which collectively form the project life cycle. Again, there are many definitions in the literature and there is no clear consensus among authors.

Schwalbe (2011, p. 70) characterizes the lifecycle as a set of project phases that define what work needs to be done, what deliverables need to be delivered and when, who is involved in which phase, and how management will review and approve the results of the work of each phase.

According to PMI (2017), the project life cycle is a collection of generally sequential project stages whose names and numbers are established by the management and control requirements of the organisation involved in the project, as well as by the nature of the project and its area in which it will be used. A project passes through the series of phases from its initiation to it closure. The phases might be broken down into functional or partial objectives, interim results or deliverables, specific milestones within the entire project's scope of work, or financial feasibility. Phases are typically time bounded with a start and end point, or control point. A life cycle can be recorded in the framework of a methodology. The project life cycle can be dictated or affected by the distinct characteristics of the organisation, industry, or technology employed. While every project has a definite beginning and a definite end, the precise deliverables and activities that happen in between will vary greatly depending on the project. Regardless of the exact activity involved, the project life cycle offers the fundamental structure for managing the project (PMI, 2017).

As these definitions show, the number of project phases is not clearly defined and is determined by the scope and needs of a particular project.

Doležal et al. (2012, p. 169–173) generalize the project life cycle into a phase model project life cycle model, which has three basic phases:

1) **Pre-project phase** – to explore the opportunity for the project and assess the feasibility of the project. Sometimes the vision, i.e., the basic idea, is included in this phase. Two main types of documents are usually prepared in this phase, namely the ***opportunity study*** and the ***feasibility study***. ***The opportunity study*** includes an answer whether it is the right time to design and implement the intended project, it has to take into account the situation in the organisation, the market situation, the expected development of the company, etc. ***The feasibility study*** is intended to show the most appropriate way to implement the project, should specify the content, the planned start and completion dates, the estimated costs, and resources of the project. This phase is generally intended to answer the strategic questions of the project – where the project is heading from, where it wants to head, what path is appropriate to take and whether the project makes sense at all to implement.

2) **Project phase** – this phase is the most comprehensive and focuses on the actual management of the project. It usually consists of ***project initiation***, ***project preparation (planning)***, ***project execution*** and ***project closure***. ***Project initiation*** is ideally a well-defined process, in line with previous events. It is needed to verify, clarify or define the project objective, required deliverables, basic staffing, etc. During ***planning***, the project team, once assembled, defines the scope of the project, creates a project management plan, identifies activities to be implemented, and creates a schedule. ***Project execution*** includes the management of individual activities. During execution, the project must be monitored, and its progress compared to the plan, corrective action taken in case of deviations, replanning and, if necessary, creating a new project baseline plan. At the ***project closure***, the deliverables are handed over, acceptance protocols are signed, invoicing is done, and the final project report is prepared. After the final phase, the project output goes into live operation, which is usually no longer part of the project. Sometimes, due to various acceptances with reservations, the project cannot be completed, and it dissolves indefinitely. This should be actively prevented, and the project should be terminated properly, or the deliverables should be declared undeliverable, and the project should be closed as an emergency.

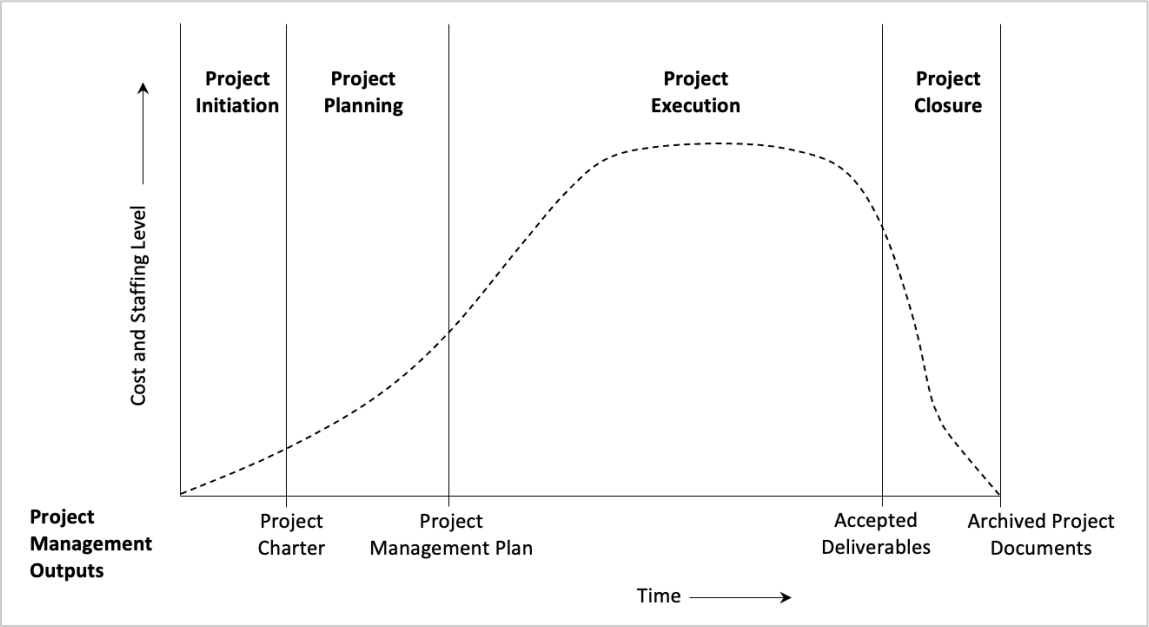
3) **Post-project phase** – in this phase, the progress of the project is analysed, good and bad experiences are identified, and the project objectives are evaluated. The evaluation is usually carried out by a different group of people than those who managed the project. The products or services are in their operational phase, but this means that team members may still have some commitment to them.

The phased model of the project lifecycle omits one important component of project management, which is risk analysis. This should be carried out throughout the life of the project, thus, since the pre-project phase already, where the first opportunities and threats should be identified.

The project's life cycle is separate from the life cycle of the product it created or altered. However, the project should consider the product's present life-cycle phase. This high-level view can offer a shared frame of reference for project comparisons even when projects differ from one another in terms of their specifics (PMI, 2017).

In conclusion, every project is always different. The complexity and size of the projects vary. Nevertheless, by PMI (2017), any project can be mapped to the following general life cycle structure *(see Figure 2 on the following page)*: starting the project, planning and preparing, carrying out the project work, and closing the project.

Figure 2 Project Life Cycle



Source: own processing according to: (PMI, 2017)

As depicted in the picture above *(Figure 2)*, the general life cycle generally shows the following characteristics: Cost and staffing levels are low at the beginning, peak as work is completed, and then quickly decline as the project comes to an end. The cost and staffing curve shown above may not be applicable to every project. For example, a project may require incurring large costs up front to get the necessary resources or it would need to start out with a full staff. The project's beginning offers the greatest potential for influencing its ultimate features without materially increasing costs, and as it moves closer to completion, that potential diminishes. In the context of the generic life cycle structure, a project manager may decide that some deliverables need to be controlled more effectively or that some deliverables must be finished before the project scope can be fully defined (PMI, 2017).

1.2.3 Role of the Project Manager

One of the possible simple definitions of project manager is that role of the project manager is the lead role for project success. It can be said that project managers play important role every time, they are leaders in every phase of a project – planning, executing, monitoring, controlling and project closing. Project manager is the person most accountable for the success of the project because they are not only in charge of managing time, cost, and quality but also integration, scope, human resources, communication, risk, and procurement (CIO, 2022, Radujkovic, Sjekavica, 2017).

A project manager is a person designated by the organisation or performing organisation to lead the team responsible for managing the project and achieving its stated objectives. This position encompasses all phases of the project, including initiating, planning, executing, controlling, and closing. Additionally, the project manager is responsible for appointing the project team,

developing the project plan, monitoring progress, evaluating objectives achieved, and the overall outcome of the project. The project manager is a managerial position with powers and responsibilities related to the project, which is usually time-limited and horizontally oriented in terms of the matrix organisational structure (Meredith, Samuel, 2017, p.11).

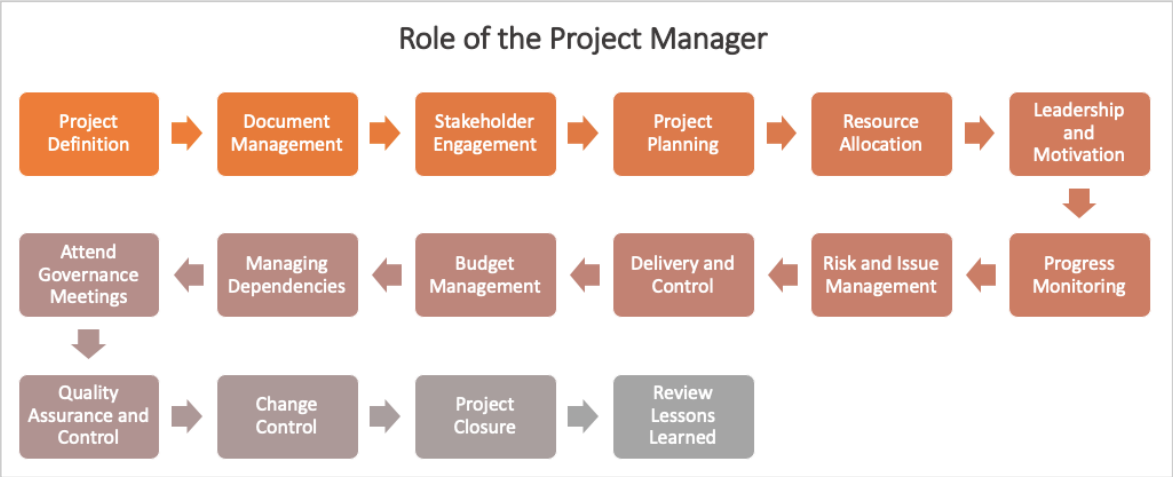
The project hierarchy typically establishes the positions of the team members, with the project sponsor being typically the company's top management. A senior management individual who is the primary coordinator and responsible for the projects' execution performs project supervision. The expert team serves as the top management's advice body. Together, they take part in the project assignment, the assessment of the work's advancement, the utilisation of the resources at hand, and the outcomes of the project's execution. Without suppliers of goods, services, and project labour, which the project manager coordinates, a project cannot be implemented. The project group managers are in charge of the work on particular sub-projects. At the beginning of the project preparation phase, the company's top management typically assembles the project team. Team members are appointed, including a project manager and other members (PMI, 2017, p. 23–24).

Project managers have significant control over the scope, team, budget, resources, and overall success or failure of a project. While technical expertise is important, effective project management also requires a range of other skills. These include a strategic approach to business, the ability to build and manage teams, resolve conflicts, and manage change. Project managers must also demonstrate leadership, inspire team members, prioritize tasks, and solve problems. Adaptability is another critical non-technical skill that is necessary for success in this profession. Good communication skills are also essential for project managers (CIO, 2022).

In conclusion, it might be deduced that project manager is the main person responsible for most of the things related to the project – for the financial and organisational management of the project, preparation of the tender documentation and implements the tender procedure. Next, responsibility for the substitutability of the project team members, including the possibility to involve external experienced persons for selected activities covered by the defined indirect costs (administration, accounting and publicity of the project). Furthermore, project manager identifies, eliminates and solves project risks, communicates with the grant provider and control authorities, prepares monitoring reports and payment requests. Supervises the eligibility of funds spent and ensures proper closure of the project, including settlement of all liabilities (CIO, 2022, PMI, 2017, p. 23–24).

The subsequent figure on the next page *(Figure 3)* illustrates the overview of the responsibilities of a project manager in performing their job duties.

Figure 3 Role of the Project Manager



Source: own processing according to: (Clayton, OnlinePMCourses, 2022)

2 Project Management Methodologies

This chapter provides an overview of three popular project management methodologies: Lean, Six Sigma, and Waterfall. The Lean methodology focuses on customer value by eliminating waste and encouraging continuous improvement. Lean principles form the basis of this methodology and include the continuous improvement process known as the "pursuit of perfection." Six Sigma uses statistical methods to improve project processes and increase productivity, while when combined with lean manufacturing, these two methods can provide powerful tools for increasing total production and efficiency. Finally, the Waterfall methodology is a traditional, linear approach to project management. This chapter aims to provide readers with a comprehensive understanding of different project management methodologies and their applications. By the end of this chapter, readers will have a solid foundation in project management methodologies, enabling them to compare those methodologies with Agile methods.

2.1 Lean Methodology

Lean thinking, a philosophy that seeks to create value for the customer by eliminating waste and continuously improving the entire value stream, begins with the belief that the customer defines value. Anything that does not contribute to satisfying the customer's needs is considered waste, and therefore, should be eliminated. Lean thinking requires a culture of continuous improvement, where everyone is engaged in finding and solving problems and there is a relentless focus on delivering value to the customer (Womack, 2018, p. 5, Managementmania, 2018).

The Toyota Production System (TPS), which is often referred to as Lean manufacturing outside of Toyota, is about creating a culture of continuous improvement by engaging every employee in the process of identifying and eliminating waste. The TPS is built on two key pillars: Just-in-Time (JIT) production and Jidoka (autonomation). JIT production means producing only what is needed, when it is needed, and in the amount that is needed. Jidoka means building in quality at every step of the production process and stopping the process immediately if a problem is detected (Liker, 2021, p. 4–5, Womack, 2018, p. 5).

Lean approach or lean philosophy is an approach to the management and operation of processes in a company, based on several basic principles. It is about the entire organisation striving for continuous improvement in all areas and avoiding unnecessary waste, and satisfying the customer's needs in the best possible way, no matter how. Lean is performing only those activities that are necessary, doing them properly and faster than others, and spending less money in the process (Bhuiyan, 2015, p. 131–132).

In conclusion, Lean thinking, the TPS, and lean methodology share the principles of eliminating waste, continuous improvement, and satisfying the customer's needs in the best possible way. Lean is not just a set of tools or practices; it is a way of thinking and a philosophy of continuous improvement that can be applied to any industry or business process, not just

manufacturing. For lean is also often used term “pull system”. The same as Agile, lean is also more about approach than a strict process. Moreover, both processes are similar in many ways and intertwine with each other. Many large companies implement lean principles, usually without much understanding by their employees. Often, they end up making some slips of paper and being “lean” enough. However, as with Agile methods, you need to understand the philosophy to get results. Likewise, a lean company should limit work in progress and be focused on finishing individual requirements the fastest way on time. Then, implement “pull system” and not to start with the analysis until a company has a priority request from its customer and until it has feedback that the previous request has been accepted (Liker, 2021, p. 4, Šochová, Kunce, 2019, p.18).

2.1.1 Lean Principles

There are five basic lean principles that build on each other and thus specify the individual processes. The last principle, which is “pursuit perfection”, essentially goes back to the beginning and thus there is continuous improvement.

The first principle is to ***Define Value***, which states that value is where lean thinking begins. Value is generated by the company that makes the goods or services; the end user is the only one who can define it. Value is only meaningful when it is described in terms of a particular good or service—often both at the same time—that satisfies the customer's needs at a certain cost and within a specific period of time (lean.org, 2022, ASME, 2022).

Mapping the Value Stream is the second principle of Lean thinking. It involves identifying all the activities needed to produce a product or service, including problem-solving, information management, and manufacturing. Value stream mapping helps organisations eliminate waste and improve efficiency by identifying the specific activities required to complete a product or service from concept to delivery to the customer. Although this can be a challenging task, it is an important step in the Lean thinking process that organisations can take to improve their processes (Marquardt, Reynolds, 2007, p.123–142, Womack, Jones, 2003, p. 19).

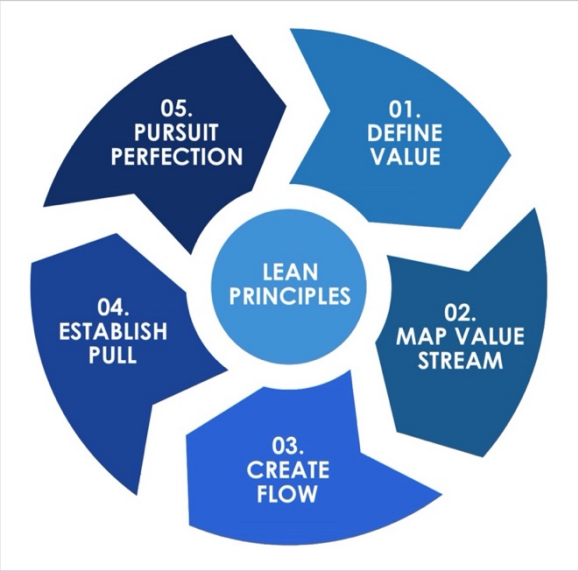
Another principle is ***Flow*** that involves ensuring the continuous movement of products through the manufacturing process, without any waste. Non-value creating processes must be eliminated, requiring a commitment to continuously improve. It should be noted that this process requires a radical change in existing thinking. However, research indicates that this will also result in significant increases in productivity and efficiency—up to or more than a 50% increase occasionally. Successful case studies demonstrate the importance of Flow in improving quality, reducing lead times, and increasing customer satisfaction (ASME, 2022, Womack, Jones, 2003, p.20).

The fourth principle of Lean thinking is customer focus or "***Pull***" production. By designing, planning, and creating exactly what the customer wants only when the customer wants it, businesses can eliminate the need for costly inventory management and accelerate time to market. "Pull" production involves producing items only when they are needed by the customer, reducing

waste and improving customer satisfaction. This approach has been successfully implemented, for example, in healthcare settings and can also be applied to the production of medical supplies and equipment, and many other industries (lean.org, 2022, *ASME*, 2022).

The first four steps are excellent places to start, but step five—making lean thinking and process improvement part of the organisational culture—might be the most important. The last principle is called ***Perfection***. Once the organisation has accurately determined value and the overall value stream, it begins to gradually create steps to define the value of specific products. Reducing costs, production times, space and errors in product offerings gets closer and closer to what the customer really wants. The four initial principles work together in a practical circle. The search for value for faster flow always reveals hidden downtime, so the deeper one goes in the process, the more obstacles that can be removed. As a result, designated product teams in direct dialogue with customers find more incentives to succeed in creating more value. As the gains mount, it is important to keep in mind that lean is not a static system and requires ongoing effort and vigilance to perfect (*ASME*, 2022, Womack, Jones, 2003, p. 25).

Figure 4 Five Lean Principles



Source: own processing according to: (ASME, 2022).

In the book "Agile Project Management Methods" by Šochová and Kunce (2019) is mentioned that Lean Software Development is based on the following principles. Firstly, it is important to remove everything that does not bring a value. In other words, "get rid of waste". Working on something that will not be used is a waste of time. Time should be invested to effective things that have sense, then, it is even much more effective. Secondly, improve a learn during the work. Everybody should learn from his mistakes and should not repeat the same mistake repeatedly. Then, the "waste" is accumulated at the end of a project more than expected. That is why regular feedback is important and helps to focus only on what really matters. Thirdly, decide as late as possible. The later a decision is made, the more information one has. For example, it does not make sense to produce stock for a warehouse just because you have a free line or programmers

at that moment. Also, deliver work as quickly as possible. The sooner something is completed, the sooner feedback is obtained that can be considered in the next iteration. Next, give the team confidence and accountability. One of the most important things is to have motivated team, that is considered as much better approach than sticking to traditional top-down structures. And finally, focus on the overall result, individual mistakes and failures are irrelevant if one learns from them. “Think big, act small, fail fast; learn rapidly” – this is the only way to ensure that the final product will be successful. The product is not just a final software, it is important to be focused on the overall impression that the product gives, pay attention to quality and overall sustainability of the system and not to create technical debt (Šochová, Kunce, 2019, p.18).

The method that applies the ideas of “lean” to software development, and that will be discussed in the following chapter, is called Kanban. It stands to somewhere between Agile methods and lean. After all, both methodologies are based on the same philosophical basis, and it is often difficult to separate them.

2.2 Six Sigma

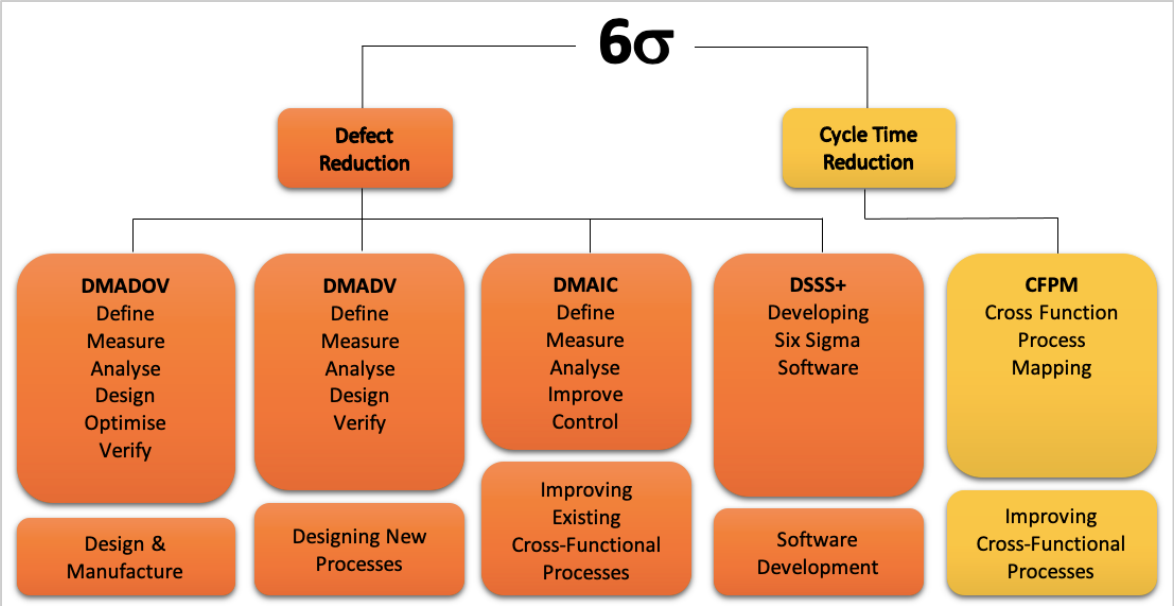
Every business owner searches for more efficient ways to manufacture their goods, reduce waste, and so increase productivity. The six-sigma approach and the lean manufacturing method are two typical methods that can increase a company's productivity. Therefore, these techniques try to assist business operations by concentrating largely on the method that raises a company's productivity by removing the procedures that often take a long time but provide little productivity. The best and most potent tools a company can have for improving its total production are created by combining these methods. The six-sigma method consists of processes and equipment that aid in applying statistics to improve processes. While the lean manufacturing approach aims to increase productivity by getting rid of waste in production systems (Kaswan, Rathi, 2019, p. 1182–1191, Henao, Sarache, 2019, p. 99–116).

Six sigma is a technique for maximising profits by ensuring customer satisfaction. Since Motorola, a cell phone manufacturer, successfully deployed it in 1986, other businesses have followed suit. This method, which uses a statistical approach, is achieved by reducing variability and offering high-quality services or products free of flaws. By decreasing inadequacies, less material is wasted, which results in efficient raw material utilisation. Statistically speaking, reaching the sixth sigma means that the obtained products or outputs have practically no defects. Six Sigma leverages data to create better solutions as it attempts to expand the organisation's capabilities and satisfy the always changing consumer wants. These are the kinds of outcomes that any organisation would like to attain (Bonome et al., 2018, p. 122–133, Beemaraj, Theni, 2018, p. 111–114).

As shown in *Figure 5* on the next page, there are many six-sigma methodology approaches. One does not need to be an expert in statistics to use the six-sigma methodology. As was already mentioned, it is explained using a statistical measure that assesses the effectiveness or quality of a service or product. Six-sigma is typically represented by the Greek symbol σ, which stands for sigma.

The standard deviation is represented by this letter. The standard deviation in statistics provides a measure of how much the data parameters deviate from the mean. If the standard deviation is low, the data values are probably closer to the mean; if it's large, the data values are probably more likely to be off the mean. This illustrates the degree of variation among the data parameters (Ikumapayi et al., 2020, p. 3275).

Figure 5 Six Sigma Methodologies



Source: own processing according to: (Ikumapayi et al., 2020, p. 3276)

By creating products with minimal variance, the six-sigma strategy aims to increase consumer satisfaction. This is essential to the production process since only products that are defect-free can be sold; those that do not meet consumer standards are discarded and sent back for remanufacturing. According to studies, when customers encounter the unfavourable effects of defective products or subpar service, they are more likely to persuade other customers to avoid using those goods or services themselves (Ikumapayi et al., 2020, p. 3276).

Lean Six Sigma is a methodical strategy that integrates the Lean and Six Sigma processes. As a result, the concepts, philosophies, and tools from both methodologies are incorporated into this approach. Lean production prioritises stages that bring value to the production by eliminating phases that are not necessary, which optimises the process and cuts down on the overall processing time. Statistical tools are used by Six Sigma to decrease process variability, which decreases manufacturing costs without sacrificing quality. The approaches of Lean and Six Sigma complement and strengthen one another. It is therefore better to combine the two methods (Siegel et al., 2019, p. 118–205).

By achieving the highest level of improvements in customer happiness, pricing, procedure speed, quality, and invested capital, the Lean Six Sigma methodology enhances productivity and raise value for shareholders. Lean Six Sigma starts with the customer, and its goal is to get rid of

anything that doesn't meet their needs. In Lean Six Sigma, it's also crucial to evaluate the consistency of products, services, and processes (Kaswan, Rathi, 2019, p. 1182–1191).

Production businesses in modern manufacturing are more focused on client pleasure while maintaining large profits, which is unquestionably a strong business model. The main goal is to organise tools in the best way possible so that they may be used in a cogent manner to ensure low production costs while also ensuring business success and low risk. Thus, combining lean and six sigma production provides the best outcomes in manufacturing a business could hope for (Ikumapayi et al., 2020, p. 3280).

2.3 Waterfall Methodology

The waterfall methodology is a project management strategy that emphasises a straight line from start to finish. This strategy, which engineers frequently employ, places a heavy emphasis on meticulous planning, thorough documentation, and methodical implementation. It is a sequential development method that moves through all project phases (such as analysis, design, development, and testing) like a waterfall, with each phase concluding entirely before the beginning of the next. At the same time, Waterfall model is the oldest and traditional model for developing software (Business.adobe, 2022, Javatpoint 2021).

The Waterfall method of system development is a well-known approach that was first described by an American computer scientist Winston W. Royce. It soon acquired support because everything goes well from the beginning to the completion of a project *(as shown in Figure 6 on the following page)*. Depending on the application, the specific processes may vary, nevertheless the fundamental steps always remain the same (Zayat, Senvar, 2020).

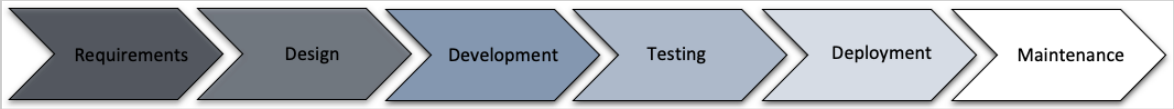
Certain individuals contend that the Waterfall methodology conforms to the adage of "measure twice, cut once." The amount and calibre of work done upfront, including the documentation of all features, variations, user stories, and user interfaces, is what determines how well the Waterfall process works. Since the bulk of the research is done up front, more precise time estimates for each requirement can be made, which can lead to a more predictable release date. It is more difficult to shift direction with a Waterfall project than it is with Agile methodology if parameters change along the way (Business.adobe, 2022).

The waterfall model outlines the essential steps in developing a product's requirements, which are then validated. A number of distinct steps, including requirements analysis, design, development/implementation, testing, and maintenance, are used to complete this process. When requirements (and risks) are predetermined and only a limited number of modifications are anticipated throughout product development, this approach performs effectively. The waterfall model might encounter numerous issues if managers, users, and developers continue to experiment with the features they want during the stages of development. Agile development was developed to effectively address this issue (Zayat, Senvar, 2020, Birgün and Çerkezoğlu, 2019).

Due to the enormous impact on the successful project management approach, the transformation from Waterfall to Agile must be assessed as a comprehensive change inside the organisation. High project restrictions and a sequential software development lifecycle are characteristics of waterfall project application. Each step must be completed and approved before moving on to the next. The scope of the project is established from the outset, and once it has been authorised, there are strict procedures to follow if the baseline agreed-upon scope needs to be changed (Popa et al., 2021).

Expected results are generally provided by the client at the start of the project in traditional project management that follows a "waterfall" process of planning and execution. The project is planned holistically, from the beginning to the end, with work packages, responsibilities, and deadlines so that it may be "worked through" in a goal- and plan-oriented manner. The basic plan is being carried out as accurately as feasible. This offers consistency and organisation, dependable resources, and planned documentation (Thesing et al., 2021, p. 747).

Figure 6 Waterfall Model



Source: own processing according to: (Birgün Çerkezoğlu, 2019)

The waterfall model is divided into several stages. The first phase is about collecting all the requirements from a client and preparing documentation. In the phase of design, a technological solution to the issues outlined in the product requirements is created by software developers, including scenarios, layouts, and data models. After the design is finished, the software development team begins to code and create the software. The project is given to the testers after the software has finished developing. The software will be tested by the testing team, and if any bugs are discovered, they will notify the developers and ensure that the bugs are resolved. They ensure that the end-to-end software is finished and a good user experience with the system. After the product has been tested, it is deployed so that real-time consumers can utilise it. Finally, the project is launched and made accessible to the clients. Clients demand a one- or two-year maintenance period since they need a team to manage any issues that arise if a bug is discovered or if they want a feature that has been somewhat enhanced in the project (Business.adobe, 2022, Javatpoint 2021).

In summary, the Waterfall methodology uses a sequential approach that relies on predefined deadlines, requirements, and outcomes. Under this approach, individual teams can work independently until specific integrations are required, and ongoing communication among teams is not always necessary. In contrast to the Agile model, Waterfall team members generally work independently and do not need to provide progress updates as frequently. Typically, the next phase of work doesn't start until the previous one has been completed (Business.adobe, 2022).

3 Agile

Agile methodologies have become increasingly popular in project management over the past two decades. This is due, in part, to the rise of software development projects with high degrees of complexity and uncertainty, which require a more flexible and adaptable approach to project management. Agile principles and practices have since spread beyond software development and into a variety of industries, including manufacturing, healthcare, and finance. In the chapter the concepts and principles of Agile will be explored, its approach in project management, the differences between Agile and traditional project management, Agile team roles, and Agile tools. The chapter begins by providing an overview of Agile, its core values, and principles. Subsequently, it will delve into how Agile approaches project management, including its iterative and incremental approach, emphasis on collaboration, and focus on delivering value to customers. There will also be a comparison of Agile project management with traditional project management, highlighting the key differences between the two approaches. Additionally, the various roles in an Agile team and the most popular Agile tools will be described. By the end of this chapter, readers should have a clear understanding of Agile principles and practices, its application in project management, and the tools and techniques that can support successful Agile projects.

3.1 Understanding Agile

The term Agile can be understood as meaning many things. Agile is a dynamic, fast, interactive, adaptive, frisky, entertaining, and quick reacting to change, and many other synonyms. Agile is a different style of life that prefers different values as real result over strict processes, a change over the pre-planned. Being Agile means to live by Agile philosophy. It brings a different company culture and mood (Cole, 2021, Šochová, Kunce, 2019, p. 13).

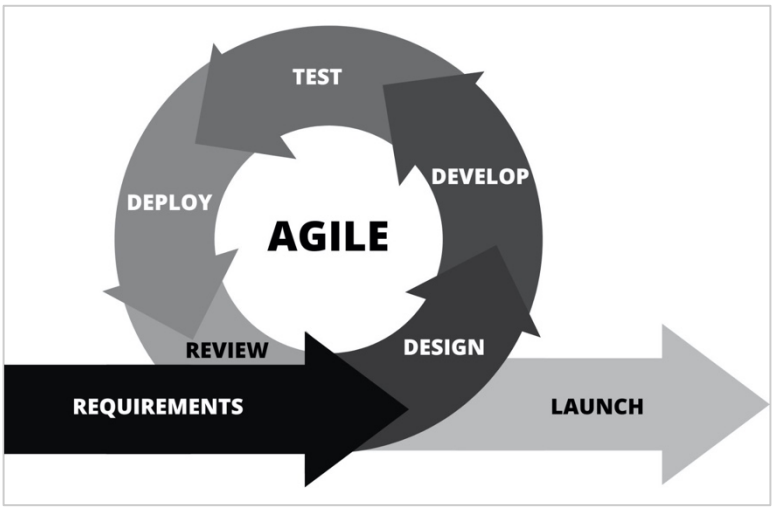
The capacity to innovate and adapt to change is Agile. It is a strategy for navigating a complex and chaotic environment and ultimately prospering in it. The name "Agile" was chosen by the writers of the Agile Manifesto to describe the entire concept because it embodied the adaptability and capacity for change that were crucial to their methodology. It basically comes down to considering how one can comprehend what is happening in the environment one is in right now, recognise the uncertainty one is experiencing, and determine how to react to it as one goes (agilealliance.org, 2022).

Consequently, Agile is about cooperation, communication, and readiness for change. Using Agile means doing the thing that makes sense at the moment and as well as possible. Agile is not a strict process, but neither a chaos. It has its clear rules. In other words, Agile defines boundaries and sets smaller playing fields within which teams can set their own rules of the game so that they can work well and be as productive, efficient and deliver a quality product in the shortest possible time. Such a team is focused on business value, or customer value. Particularly, focusing on how to optimize the functionality so that the customer is maximally satisfied and gets what he really needs and can use for the money spent (Cole, 2021, Šochová, Kunce, 2019, p. 13).

Originally, the Agile methodology was established to encourage change and embrace early input in the software development life cycle, in order to continuously satisfy client needs. (Popa et al., 2021). During Agile life cycle *(shown in Figure 7)* the team plans, develops, reviews, and deploys updates to the product's functionality for each iteration. Stakeholders may request new priorities or requirements as they carry out acceptance testing and see the finished product. The Product Owner adds that feedback to the product backlog where it will be prioritised along with other features and tasks. Then, the procedure is repeated (Blake, S., EasyAgile, 2021).

Agile software development is a collection of approaches in which several teams with various backgrounds work together to create solutions. Coordination between the software engineering team and the customers promotes quick response to customer orders and a high level of adaption to any new requirements. The belief that changes will positively affect the creation of goods makes Agile's support for a culture of change a key component. It also embraces the concept of responding to changes by using a clear model (Cluster, 2019, Zayat, Senvar, 2020).

Figure 7 Agile Life Cycle



Source: own processing according to: (Blake, S., EasyAgile, 2021)

The demand of today's markets, which require innovative and high-quality software products in a relatively short amount of time, was the natural answer to Agile. Two of the most pressing demands for the majority of today's business and technology workplaces are met by Agile software development: the need for an innovative, dynamic approach to project management and the desire to design workplaces in a way that ensures effective communication between individuals. Developers, clients, and sponsors can overcome obstacles, establish priorities, and look into alternative routes through active communication. This can reduce expenses, expedite the procedure, and increase manoeuvrability (Zayat, Senvar, 2020).

In 2001, seventeen Agile leaders in Utah wrote the mentioned Agile Manifesto, which is documentation based on four key values – individuals and interactions over processes and tools, working software over exhaustive documentation, collaboration with the customer over contract negotiation, and responding to change over sticking to the plan. A manifesto outlining values and

12 principles for Agile software development was developed to address the increasingly diverse requirements of customers. Agile proved to be a fitting solution for software developers in this regard. Moreover, unlike traditional development approaches, Agile does not mandate extensive documentation of the product. Instead, it prioritizes customer satisfaction, fostering a sense of trust and collaboration between the development team and the customer (agilemanifesto.org).

The most important of the 12 Agile Manifesto principles is to meet the customer's needs to deliver software that adds value to the customer on a continuous and timely basis. Another principle is that changes in requirements are welcomed throughout development, even in its later stages. The third principle of Agile management is that Agile processes encourage change, which in turn leads to an increase in the customer's competitiveness. Agile principles also include delivering working software, in shorter periods of weeks or months, or collaboration between development and business people on a daily basis throughout the life of a project. The following principle is building projects around individuals who are motivated and given the right environment and support to get the job done well. Another principle is the importance of face-to-face conversation, which is the most effective communication channel within the development team and beyond. One of the principles is also working software, which is a major criterion for progress. Next principle is the promotion of sustainable development and a constant pace, which should be maintained at all times, not only by developers but also by sponsors and individual users. Constant attention to good design and technical excellence that increases Agility is another principle. The tenth principle is that simplicity, which maximises the amount of work not done, is essential. The penultimate principle is the initiative of self-organising teams from which the best requirements, designs and architectures will subsequently emerge. The twelfth and final principle is the regular retrospective of the team to modify its habits and behaviours to become more effective (agilemanifesto.org).

As Šochová and Kunce (2019, p. 13) state, being Agile is not as easy. It’s not about to pass an Agile course and then immediately become Agile. No Agile certificate might help one in understanding of Agile, however, it can help one to start the process of change. Nevertheless, a person needs to be Agile, think agilely, behave agilely.

3.2 Agile Approach in Project Management

Some projects are characterised by a high degree of uncertainty e.g., the development of new technology, products, or other results, often of an intangible nature. There is a lack of information for reliable estimates, the context of the project or the project itself is affected by many changes, often in the terms of reference, etc. This makes it difficult to draw up a workable project plan. The solution to this situation is “Agile project management”, which has emerged from the field of software development and design. Agile project management was developed as a reaction to traditional project management approaches, where it is necessary to respond to changes in technology and ICT, e.g., R&D is characterized by a high level of risk or uncertainty and the inability to specify the target product at the beginning of the project (Petrtyl, Skalický & Vacek, 2012, p. 2).

In the context of project management and the corporate environment, the term is understood as a nimble and flexible management approach that allows projects and companies to respond easily to change, so that Agile methods are characterized by the fact that they allow to survive in an atmosphere of constant change and enabling success to be achieved. As was mentioned in the previous chapter, the phrase "being Agile" is most commonly used in software development companies and projects, where developers have realized the need for a change in project management, a change in the application of "classic" project management tools and techniques. Therefore, a relatively separate group of methodologies called Agile project management has been singled out (Popa et al., 2021, Chow & Cao, 2008, p. 961–971).

Agile project management is an iterative method for delivering projects at all stages of their life cycles. Multiple iterations or incremental steps toward the completion of a project make up iterative or Agile life cycles. Since the benefit of iteration is that you may alter as you go along rather than following a linear path, iterative approaches are widely employed in software development projects to promote velocity and adaptability. Release of benefits throughout the process as opposed to only at the conclusion is one of the goals of an Agile or iterative strategy. Agile initiatives should fundamentally exhibit the principles and behaviours of cooperation, adaptability, trust, and empowerment (Association for Project Management, 2022).

According to Svozilová (2011, p. 362), consultant and manager with extensive experience in project and process management, the main characteristics of Agile project management can be summarized in five points. The first characteristic of Agile project management is continuous innovation, where the subject matter resulting from the project is assumed to meet current customer needs in turbulent conditions market environment influences. Next, continuous adaptation of the project goal to be able to meet the future needs of the project customer. The third point is accelerated introduction of the product on the market. Another principle is increasing the adaptability of processes and people so that they can respond to changing requirements for the project subject as well as to ongoing changes in the environment. And the last point, ensuring the reliability of the project deliverables produced to support the success and profitability of the project contracting authority.

Agile project management, with its roots in software development, is being adopted in an increasing variety of industries (Sutherland & Sutherland, 2019). According to the 15th State of Agile Report by Digital.ai (2022), 94% of software organisations claim to have used Agile development at least occasionally. In the software sector, Agile development expanded in 2021, increasing from 37% to 86% of teams. This is an increase of 232%. The adoption rate doubled annually in non-IT industries, which also saw significant growth. Agile was used by 63% of IT, 29% of Operations, 17% of Security, and 16% of HR. Moreover, 43% of marketers utilise Agile in some capacity (Digital.ai, 2022).

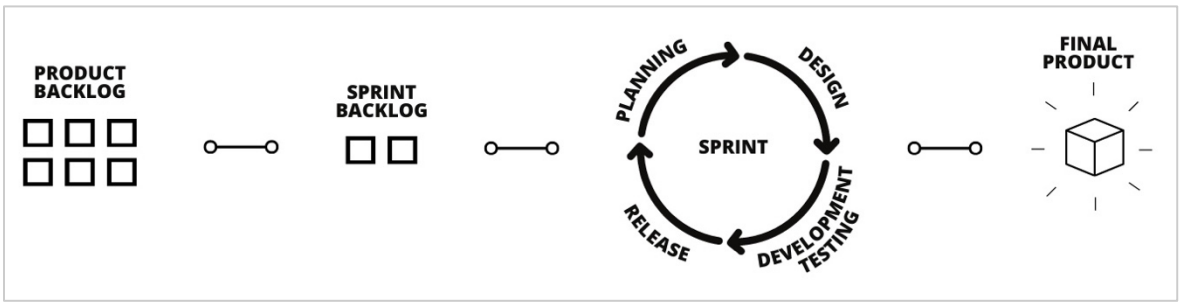
Comprehensive advanced planning and the sequential, precise "implementation" of a plan are not the main objectives of Agile methodologies like Scrum or Kanban. As opposed to this, a project team develops a solution separately and quickly cycles through interim outcomes with the

customer. The customer or user of the project outcome expresses general requirements but is unable to explain these in detail in the early project phases, which is one justification for using this method (Sutherland & Sutherland, 2019).

The steps necessary to accomplish the objectives might not be evident. The Agile approach still establishes objectives or a vision for the project, but it does so with less commitment, at a lower level of detail, and over a shorter period of time (for example, two to four weeks). As expectations for the outcome are concretised during project implementation, flexibility in the case of modification requests is more crucial than strict adherence to the original plan. As opposed to the traditional waterfall method, the project process is not linear in terms of subsequent phases. Instead, a test-driven technique that involves several iterations may be employed to achieve the desired outcome. Agile project-management techniques give projects flexibility, allowing businesses to respond swiftly to shifting client demands (Wysocki, 2019, p. 381–382, Preußig, 2018, p. 63–64).

The basic premises of Agile project management are dividing a project into short intervals, as opposed to a simple linear waterfall model. Scrum methodology uses the term “sprint”, constant interaction with the project’s customer, gradual refinement and possible adjustments of the deliverable and informality, where personal interaction is preferred. Each of them functions as a miniature project, complete with a backlog and the design, implementation, testing, and deployment phases all falling under a predetermined scope of work (Doležal, Máchal & Lacko, 2012, p. 190, *Altextsoft*, p.6).

Figure 8 Agile Development Cycle



Source: own processing according to: (Altextsoft, p.7)

A potentially shippable product increment is delivered at the conclusion of each sprint. As a result, the project gradually grows as new features are added to the product with each iteration. Delivering a possibly defective product is far less likely when the features are validated early in the development process (Altextsoft.com, p.7).

3.3 Agile Project Management vs Traditional Project Management

Even with Agile project management, the key principles remain the same, the difference can be seen in the level of detail required to proceed. For example, if we require in a "classic" project a fixed scope and content of the project in the form of a WBS or SOW, in an Agile project management we only need to know the desired target state. The project team will always plan the detailed plan of sub deliverables and timeline for the upcoming iteration, not until the end of the project. At the end of each iteration, the deliverable is handed over, the progress to date is evaluated, and the content of the next section of the project is determined. Over time, the principles of Agile methods have penetrated and continue to penetrate beyond the realm of software design, thus they are also used for projects in other areas. The growing popularity of Agile project management is evidenced, besides other things, by the fact that PMI is currently introducing a special certification for Agile project managers, different from the established PMP level (Doležal, Máchal, Lacko, 2012, p. 191, 360PMO Project Management Consulting Inc., 2018).

IT Manager, Dr. Moran (2015), states that Agile project management also turns the traditional project triangle (of time, cost and scope) and its newly formed by:

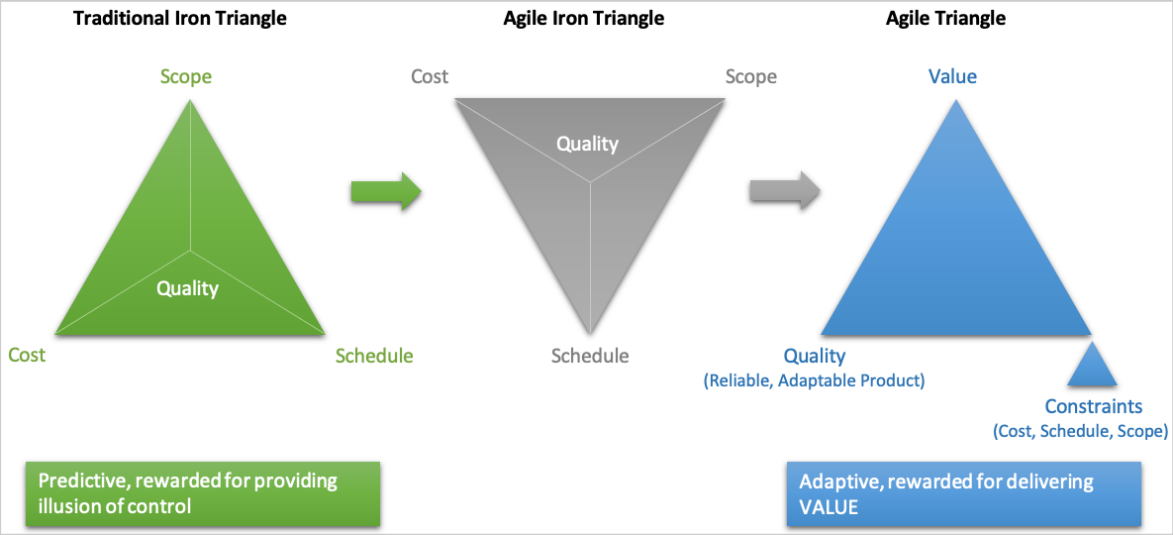
- **Value** – to develop/build a project item that has the protentional to be marketable or meet the internal business needs of the client.
- **Quality** – to develop/build a project item that is reliable and able to adapt to the needs of the project customer.
- **Constraints** – the set of conditions that must be met in order to fulfil the above Value and Quality objective.

As was mentioned in the first chapter, traditional iron triangle is focused on three elements, which are scope, cost, and schedule/time. Agile triangle, on the other side, emphasises value, quality, and constraints, which are cost, schedule, and scope. For example, an organisation needs a software for managing a school's library system, through which students will borrow and return books. That establishes the scope. Cost is essentially the amount needed to finish the entire project. A schedule is a list of deadlines for doing particular tasks (Moran, 2015, p. 72–73).

Despite including all of these components, the Agile triangle does not regard them as the project's primary goal. They are never completely abandoned, but they are also not the main focus. Value and quality are the two main focuses of Agile triangle. Value identifies the key features that customers value most in software. In actuality, it encompasses the interior scope. Quality is the ability to satisfy the expectations of the consumer while producing a dependable product. The iron triangle states that scope, cost, and schedule determine quality. On the other hand, quality in the Agile triangle allows for flexibility in terms of cost, schedule, and scope. Scope, cost, and schedule are thus included in constraints as part of the Agile Triangle, allowing

for a more flexible and adaptable approach throughout the duration of the project (Yusuf Adanur, 2021).

Figure 9 The Evolution to an Agile Triangle



Source: own processing according to: (360PMO Project Management Consulting Inc., 2018)

As shown in the picture above *(Figure 9)*, scope, schedule, and costs make up the left-most triangle of the classic iron triangle of project management. Although many managers make an effort to control all three dimensions, in many situations scope was the main factor (due to the fallacious belief that scope was understood early in the project) and cost and schedule varied.

The second triangle depicts an early view of measuring Agile development in which the scope was left open-ended, and the timetable was time-boxed, i.e., time was utilised as a fixed constraint. Unfortunately, this second triangle still complies with the existing guidelines for the iron triangle. Success is still defined by compliance to cost, schedule, and scope in many firms and is continued by research organisations like the Standish Group. If success is defined as following the plan, then how will Agile projects that continuously adapt be regarded successful? This raises the third triangle, the Agile Triangle. In the Agile Triangle, value (to the customer), quality (needed to offer consistent value to the consumer), and constraints (scope, schedule, and cost) are the measures. Although they are not the project's objective, constraints are still crucial to project parameters. The goal is the value and constraints may need to be changed as the project progresses to increase customer value. Even while the schedule may still be a fixed constraint, the scope may be altered to maximise value while maintaining the schedule constraint. If an organisation wants adaptability, it must reward it. Organisations can fill this need by adjusting constraints to achieve value or quality objectives. To summarise:

- **Value goal:** Build a marketable product.
- **Quality goal:** Build a dependable, adaptable product.
- **Constraint goal:** Reach value and quality goals within allowable constraints.

Agile project management can be used by organisations to manage marketing projects, launch new products, manage regional or multinational expansion and much more. However, what is clear is that Agile methodologies may not always be the most appropriate approach. In practice, several project-specific parameters need to be considered. Nevertheless, a combination of both, traditional and Agile project management, may also be appropriate (Highsmith, 2013, p. 87).

Therefore, Agile project management considers time and resources as the fixed variables and scope as the variable, which is adapted to the customer's priorities. In order to compare the two approaches of the so-called, traditional methodologies and Agile methodologies, a comparison is made in the following table where the main differences are presented.

Table 1 Comparison of Traditional and Agile Methodologies

Viewpoint	Traditional Methodologies	Agile Methodologies
Content of the Methodology	Processes, they focus on explicit knowledge and view people as a secondary factor	Practices, focus on "tacit" knowledge, understand people as key success factors
Detail Methodologies	Processes and activities are described in great detail	Focusing on activities that create value and the elimination of activities that do not bring value
Quality	Focus on quality processes that will lead to a quality outcome	Focus on customer value and high product quality
Predictability	Assumes predictability of the future, emphasis on anticipation (gathering requirements in advance, forward planning)	Assumes unpredictability of the future, emphasis on adaptation to change (incremental requirements gathering, planning for iteration)
Changes	Changes are subject to change management, efforts to minimise changes	Striving to enable and exploit change, allow customers to rethink their requirements considering new knowledge
Definability of the development process software	Software development is a defined process, it is possible to repeat it without a problem	Software development is an empirical process, it cannot be consistently repeated, but requires constant monitoring and adaptation
Value for Customer	The assumption that good processes lead to good results; too focused on the processes themselves, not on the results for the customer	Top priority is to satisfy the customer
Customer Participation in the Project	In the initial and final stages only; after signing the management requirements specification document, the	Shifting control from the team to the customer; the customer is the controlling entity throughout the

	team takes over of technology staff	project, the customer can change function priorities at each iteration
Scope of the Solution	Developers try to build into the system all the features that the customer might need in the future	Required functions only, Requirement for minimisation
Relationship between Customer and Developer	Secured by contract, distrust	Trust and Cooperation
Human Factor	Secondary; documentary-oriented processes try to cast people in the role of interchangeable parts	Primary; uses individuality and strengths of people
Qualification of People	Just standard individuals	Emphasis on people's skills, knowledge, and abilities
Specialists x Generalists	The requirement for close specialisation of people	Requirement for knowledge integration and constant collaboration; knowledge sharing in a team; team-based problem solving; generalists rather than specialists
Management Method	The traditional way of management is shaped on based on distrust, directive management, controls	Leadership and Collaboration. is formed on trust and respect
Importance of Programming in Software Development	Emphasis and value are placed on architecture, requirements and design, coding and testing are seen as low "design" value activities	Emphasis on programming as an activity that brings value
Simplicity	Rather complex solution that tries to encompass future requirements	Emphasis on simple solutions, no building in future requirements
Simple x Complex rules	Methodologies try to describe everything the development team may encounter	Contain generative rules – the minimum set of things you must do in all situations
Modelling	Great emphasis on modelling, especially advance modelling – big design in front of, then freeze the request	Agile modelling – modelling is not about the model itself, but the act of modelling; the purpose of modelling is communication
Form of communication	Predominantly written	Emphasis on face-to-face communication

Documentation	Extensive documentation	It is not the documentation that is important, but the understanding
Mode of Development	Rather waterfall, possibly iterative and incremental with long iterations	Incremental development with very short iterations
Economy	Sources tend to be a variable that usually increases	Always striving to realise the highest value of the money given; the goal is value for the customer, not a perfect system; value is the combination of product features that meet customer needs at a particular time and for a particular price

Source: own processing according to: (Buchalcevová, p. 51–52)

Some values in the Agile manifesto are prioritised over others. Relationships with users and consumers are more crucial than rigidly following the procedure and the tools themselves. On the other hand, the traditional approach to software development places more emphasis on the tools and process itself. The working software, not the process documentation, is what counts in the Agile manifesto. In the conventional approach to software development, software documentation is crucial. Agile processes prioritise customer collaboration over concentrating on contract negotiations. Traditional software development, however, strictly adheres to contracts and forbids any modification. Therefore, client collaboration is not a frequent practise in traditional software development. The Agile Manifesto encourages changing as projects progress. On the other hand, the traditional software development method closely adheres to the development plan. The traditional software development methodology does not like change. As stated in the interdependence statement, Agile processes foresee unpredictability and respond pro-actively and appropriately through iterations. Uncertainty is unacceptable since change is not anticipated in traditional software development (Yusuf Adanur, 2021).

3.4 Agile Team Roles

Agile teams are being used by more and more organisations nowadays. A "team" in the Agile definition is a small number of individuals who are assigned to the same project or effort and work together almost exclusively full-time. The fundamental idea behind an Agile team is to create a group of people who work together towards a shared goal while being flexible to meet changing customer needs. These individuals practice a form of collaborative leadership, are self-motivated and organised, which distinguishes them from traditional teams. The idea of a team implies shared responsibility; whether the results are good or negative, they ought to be given to the entire team, instead of being given to a specific person. The team is anticipated to have all required skills, whether technical (programming, designing, testing) or managerial (domain knowledge, decision making ability). Nevertheless, results are more important than roles and responsibilities. Agile

teams are designed with the premise that they can operate more quickly and adaptably than traditional project groups. Agile teams are made to work incrementally, finishing one aspect of the project at a time, which boosts productivity and lowers risk (agilealliance.org, 2022, Vasiliauskas, teamhood.com, 2023).

As mentioned in the previous chapters, Scrum is currently a very well-liked Agile methodology. A Product Owner, the Development Team, and a Scrum Master make up the ***Scrum team***. Scrum teams are the same as Agile team cross-functional and self-organising, they make their own decisions on how to complete their tasks rather than following instructions from outside the team. Cross-functional teams has all the skills required to complete the assignment without relying on outside sources. The Scrum team structure is created to maximise adaptability, creativity, and productivity. Scrum teams produce things incrementally and iteratively, maximising the chance for feedback. Thus, an Agile team can also be called a scrum team, however, the primary distinction between Agile and Scrum is that Scrum is a particular Agile methodology that is employed to facilitate a project, whereas Agile is a project management philosophy that makes use of a fundamental set of values or principles (Stray, Memon, Paruch, 2020, Schwaber, Sutherland, 2017, p.6).

One of the Agile team roles is an ***Agile coach*** that is frequently employed to help teams and organisations in adopting and benefiting from Agile methods. The position is also known as a DevOps coach, scrum coach, Kanban coach, or lean coach. Agile coaches can be external consultants hired by a corporation or inside staff members who take on coaching tasks. The function of an Agile coach began to diverge from that of a Scrum Master, becoming more closely linked to leadership, teamwork, and performance than the Scrum framework. An Agile coach strives to improve collaboration, information sharing, and problem resolution while boosting motivation. Similar to a typical team coach, the primary objective of an Agile coach is to improve teams' capacities for achieving greater levels of accomplishment. In summary, effort, performance tactics, knowledge, and skills are all addressed by Agile coaches. By boosting team members' motivation to accomplish goals, the Agile coaches have an impact on their effort. The Agile coaches coach for improving work processes while focusing on performance techniques. They assist the team in coming to consensus on the team's rules and improvement strategies, and they increase the team's understanding of its members' responsibilities and work procedures, thereby lowering stress (Stray, Memon, Paruch, 2020, Stray, Tkalich, Moe, 2021, DeRue, Barnes, Morgeson, 2010).

The Scrum framework suggests that the ***Scrum Master*** assume team leadership responsibilities. The Scrum Master is seen as a facilitator of the Scrum process who makes it possible for a team to work autonomously and cross-functionally. The Scrum Master help the team in putting Agile techniques into practice. The Scrum Master role is support and guide the team through the process, and also protect the team from outside interference. In comparison to a project manager, to produce the desired outcome, project managers assign tasks, keep track of deadlines, manage risks, and supervise the team. The Scrum Master, on the other hand, is completely concerned with using Scrum techniques. They aid in the team's understanding of how things work but have little influence over how tasks are assigned, deadlines are set, or other

relevant issues. A qualified Scrum Master helps teams estimate projects and boost productivity by removing roadblocks. To help coordinate the process in an efficient and visible manner, they may utilise an Agile board like Kanban. The Scrum Master also facilitates sprint planning and scrum meetings. Along with motivating the team, the Scrum Master holds the group together and works to enhance its dynamics. They provide assistance to every project component that need it at a particular time. Instead of performing tasks for the team, the Scrum Master's role is to empower the team members (Vasiliauskas, teamhood.com, 2023, Schwaber, Sutherland, 2017, p.7).

Another important role in an Agile team is the ***Product Owner***. Product Owner is in charge of maximising the value of the final product that the development team produces. The used methods can differ greatly amongst organisations, Scrum Teams, and people. The Product Owner is the lone person in charge of overseeing the product backlog – a list in order of everything known to be required in the product. Product backlog management entails clearly expressing product backlog items, organising the product backlog items to best achieve goals and missions, maximising the value of the work the development team completes, ensuring that the product backlog is visible, transparent, and clear to all, and indicating what the Scrum team will work on next, as well as making sure that the development team fully comprehends the product backlog items. These tasks may be carried out by the Product Owner or have the development team to do it. The Product Owner is still responsible, though. Product Owners are individuals, not groups. The Product Owner must be respected within the organisation for success. The order and content of the product backlog reflect the Product Owner's choices. Nobody has the right to compel the development team to work under a different set of requirements. However, in an Agile team, there may be a person who assists the actual Product Owner in managing the backlog, called a ***Proxy Product Owner***. A Product Owner Proxy is responsible for helping the Product Owner define, maintain, and track requirements during the development process. In some situations where the Product Owner is unavailable, it may be argued that the proxy Product Owners serve as a conduit for information between the Product Owner and stakeholders, acting as their business advisors (Schwaber, Sutherland, 2017, p.6, Latre A., scrum.org, 2017).

3.5 Agile Tools

Agile tools are project management tools created to support Agile methodology. Agile tools are used to help achieve the key aims of Agile project management as adapting and organising solutions and requirements to enable cross-functional teams to collaborate, organise, and produce the best solutions. Project planning, development, testing, and the creation of a long-term product roadmap are all made simpler by Agile project management tools. Efficient Agile project management software improves customer satisfaction, financial performance, and team effectiveness (Williams, C., acelo.com, 2021).

Although there are many Agility supporting tools, this chapter briefly describes the best known and most used Agile tools. One of the known virtual tools supporting Agile is ***Mural***. Innovative teams can explore difficulties or challenges using Mural to visually interact and share

ideas. Thanks to Mural, Scrum for team project management is simple to implement, teach, and apply. Both, Agile project management and general project management, are supported by Mural, which offers a place for design thinking exercises and sprint planning/reviews. Mural enables innovative teams to draw straightforward yet thorough diagrams and brainstorm in a virtual environment. Mural resembles a "thinking canvas" in many aspects, which teams use to arrange their ideas into lists, flowcharts, frameworks, or drawings (Chin, D., mural.co, 2022, innovationtraining.org, 2022).

Another known Agile software tool is ***Jira*** that aids in the process of managing projects and requirements. Atlassian Jira is a set of Agile work management results that provides adaptable and user-friendly solutions for task management and employee monitoring. Jira is a tool that facilitates collaboration among teams, enabling them to work together more effectively across a variety of contexts. From Agile software development and customer service to start-ups and established companies, Jira is used by teams to streamline their workflows and achieve their objectives more efficiently. UI/UX designers, development teams, and IT professionals are the main target audiences for Atlassian Jira. It includes completely adaptable Kanban scrum boards that teams can use to fit into the overall workflow of a company. These boards can be used as a quick method to see all of the tasks involved in a sprint. In summary, Jira is a tool that helps teams with various aspects of work management, such as planning, tracking, reporting, and assignment. For organisations that require more extensive Agile planning capabilities, there's ***Jira Align***, an enterprise-scale software solution that connects work across teams and projects. Additionally, Jira Align enables data organisation and connection through the development of real-time reports covering every facet of a whole organisation. This implies that each profile involved in the process will receive the information it needs. Jira and Jira Align are different in that Jira links teams to other teams, whereas Jira connects teams to the business. Both are arguably among the most well-known Agile project management tools ever, according to Atlassian, Jira is the tool number one for Agile teams (Montoya, G., *blog.deiser.com*, 2022, attlasian.com, 2023).

Many teams also use Microsoft tools supporting Agile such as MS Teams, MS White Board, MS PowerPoint, MS Excel, or MS Planner. ***Microsoft Teams*** an application that offers a workspace for real-time collaboration, meetings, file and app sharing, and chat. Within the Microsoft Teams environment, teams may easily adopt the Agile project management methodology. Teams who adopt scrum will find they can get a lot out of the solution to fulfil their goals, the Scrum Master, and the scrum team both have a lot of capability to keep things organised and on track. Microsoft Teams completely covers Agile teams' task management needs because dependencies and alerts may be easily customised for situations where individuals have to adjust to an unexpected change. Numerous organisations all around the world utilise Microsoft Teams to improve collaboration and support teams in keeping all their important documents and processes in one spot, including invoicing. Most project management software does not come with an invoicing capability; however, more and more Agile teams are going in this direction toward more all-in-one solutions (Riaz, Q., *psohub.com*, 2022).

A brief description of other popular Agile tools is illustrated in the table below (*Table 2*):

Table 2 Agile Tools

AGILE TOOL	DESCRIPTION
MS WHITE BOARD	A revolutionary, free-form digital canvas that enables team members to communicate anywhere in real-time. A limitless canvas made for pen, touch, and keyboard allows teams' ideas room to develop.
MS POWER POINT	A comprehensive presentation software included in the Microsoft Office suite, which also includes popular office productivity tools such as Word and Excel. The software allows users to create multimedia-rich presentations using a series of slides, making it an effective tool for conveying complex information in a visually engaging manner.
MS EXCEL	A Microsoft spreadsheet tool that makes use of tables to store and analyse statistical and numerical data through functions and algorithms. Since the release of version 5 in 1993, it has dominated the market.
MS PLANNER	A lightweight, web-based, mobile app that is included with the majority of Office 365 for business subscriptions. Teams may make plans, assign tasks, discuss about assignments, and view charts of their team's progress with Planner.
KANBANIZE	An online Portfolio Kanban tool for Agile project management where teams may visualise projects, monitor progress, and maximise workflow effectiveness to deliver value to the market more quickly. Teams can track and distribute the work inside an organisation with just one management board. Kanbanize was created with scalability in mind; the Agile tool integrates Kanban-style elements meant to aid businesses in achieving organisational transparency and accelerating business Agility.
TRELLO	A free online list-making tool that uses the Kanban method. Trello is a well-liked, straightforward, and user-friendly collaboration platform that organises projects into boards, and lists what is being worked on, who is working on it, and what stage of a process it is in.
ACTIVECOLLAB	An integrated, user-friendly project management and collaboration platform for creative professionals that integrates task management, time tracking, and billing. Due to its reporting and budgeting tools, Active Collab is the perfect Agile project management application for small businesses, particularly ecommerces. ActiveCollab provides self-hosting licences or subscription plans based in the cloud.
PIVOTAL TRACKER	An easy-to-use, cloud-based Agile project management tool created to promote team collaboration and track development over the course of the whole project lifecycle. Based on the team's continuing performance, Pivotal Tracker assists software development teams in setting reasonable expectations for when work might be completed. Additionally, team

	members can use Scrum points to assign a complexity rating to each assignment.
LEANKIT	A Kanban software that offers improved task visualisation and more flexibility for ongoing process development. This tool for Agile project management is especially designed for teams who must collaborate remotely. LeanKit makes it easier for those working in application development, DevOps, and IT operations to apply lean concepts and scale Agility across the organisation. Furthermore, it works well for cross-functional teams that use the scrum or Kanban methodologies.

Source: own processing according to: (Downs,J., justinmind.com, 2020, microsoft.com, 2022, simplesheets.co, 2022)

The 16th annual State of Agile report, the latest update annual report for the year 2022 by Digital.ai, shows data that Jira is unquestionably the most widely used Agile tool. Atlassian Jira is used by two-thirds of respondents to manage Agile projects, while Mural by over 2 in 5 and Microsoft Excel is used by a third of people. The State of Agile report is the longest ongoing annual series that provides an overview of Agile methods and practices and examines the impact on people, processes, and tools by sharing the experiences of more than 3,000 people in the Agile community. The 16th State of Agile Report discusses significant trends and problems with Agile adoption and practice for businesses as they expand Agile beyond software development teams to the full enterprise (Digital.ai, 2023).

4 Agile Methodologies

The software development industry has undergone significant changes over the past few decades, with the introduction of various methodologies and frameworks to improve the efficiency and quality of software development projects. Agile methodology is one such framework that has gained widespread popularity in recent years due to its focus on flexibility, adaptability, and customer satisfaction, not only in the software development industry. The aim of the last chapter of the theoretical part is to provide the reader with a general overview of the most well-known Agile methodologies. This chapter provides an overview of Agile methodologies, including popular methods and techniques, the process of transforming to Agile, and detailed explanations of the most popular Agile methods, namely Kanban and Scrum methodologies. Additionally, subchapters explore the specific events and artifacts associated with the Scrum methodology and the values underlying the Kanban approach. Through this exploration, readers will gain a deeper understanding of Agile methodologies and their potential to improve development processes.

4.1 Agile Methods and Techniques

In the early years of 2001, software development professionals met in Snowbird, Utah, to discuss alternative software development methodologies, and during this conference, the concept of Agile was developed. Twelve principles and a set of values were used to form the Agile software development manifesto. Agile was exactly what developers required to handle customers' increasingly diverse requests. As was previously mentioned, Agile places less emphasis on the product's documentation system and more on the customer, which fosters trust between the creating team and the client. This could be what causes the Agile model's biggest flaw. The fact that Agile projects are produced progressively may be considered as the biggest distinction between Agile and other traditional methodologies (agilemanifesto.org, Cluster, 2019).

As was mentioned, Agile is an iterative process with a time limit, which promotes quick and adaptable responses to changes. Agile's goal is to ensure successful customer and development team cooperation by promoting customer participation at every stage of the project, from prioritising user stories to iteration planning and iteration reviews. Thus, Agile is built on incremental and iterative development, in which customer and self-organising, cross-functional teams work together to produce requirements and solutions. This demonstrates a great degree of openness between the client and the development team. Agile emphasises business values since teams can determine what is crucial for the customer's company as they participate in the prioritisation of the features. Agile methodologies such as Scrum or Kanban might make it easier to accomplish Agile goals. Scrum and Kanban, which are based on Agile principles, are two most well-known Agile techniques that have been used successfully by thousands of businesses since the turn of the century. The application of Agile principles uses a variety of frameworks. A framework is a strategy or set of techniques and principles used to execute Agile and uphold its principles (Cole, 2021, Zayat, Senvar, 2020).

There are many different Agile frameworks available. The most popular and widely adopted frameworks are Scrum, Kanban and Lean. The Lean method is described in the previous chapter *(Chapter 2.1)*. The most popular Agile methods, namely Scrum and Kanban, are described separately in more detail in the following chapter, while other known Agile frameworks are described more briefly in the table below *(Table 3)*.

Table 3 Agile Frameworks

AGILE FRAMEWORK	DESCRIPTION
Extreme Programming (XP)	A standard Agile development approach focusing on software engineering that centres on the notion of finding "the simplest thing that will work" without giving undue consideration to the long-term product view. XP emphasizes principles like communication, simplicity, feedback, courage, and respect and places the highest priority on customer satisfaction. It is also very team-oriented, which implies that everyone in the development team, not just the owner or team leader, is responsible for the project's successful completion. With this practice, developers are encouraged to accept adjustments to customer requirements, even if they come in the middle of the development cycle. XP is best practiced in small teams.
Nexus	Agile framework that is applied to scaled Agile projects with three to nine Scrum development teams, each with five to nine people, and a single shared product backlog that is used by all of the teams. The Nexus framework overlaying and extends the current Scrum Agile methodology. It functions as Scrum's "eco-skeleton." The primary distinction between Nexus and Scrum is the addition of an integration team, which is responsible for resolving dependencies and integration issues amongst the teams. Nexus is the most simple and lean framework in comparison to the other frameworks, allowing for a lot of flexibility.
Scaled Agile Framework (SAFe)	SAFe is an interactive software framework that makes it possible to use Scrum and Lean-Agile techniques at massive corporations. SAFe helps organisations manage issues like funding, product roadmaps, and change management. Additionally, it explains how to measure success in achieving their objectives using Lean concepts. SAFe is nearly comparable at the team level with Scrum. SAFe is quite directive in the sense that it instructs organisations on what to do. Additionally, SAFe is rather simple to implement and is frequently fully adopted by enterprises, its prescriptiveness is one of the strong points; however, it also has a drawback in that it prevents teams from exercising as much flexibility in process decisions, which makes them less adaptive. However, it is one of the most extensively used scaled Agile frameworks, has been shown to be effective, and certified training is easily accessible.

Large-Scale Scrum (LeSS)	After applying Scrum's ideas and guidelines to significant projects, Bas Vodde and Craig Larman began to design the LeSS framework in 2005. Their objective was to successfully construct massive projects within the bounds of Scrum. LeSS comes in two varieties – ***Basic LeSS*** that involves two to eight teams of eight people that each working on the same product development, and up to 2,000 workers developing the same product is ***LeSS Huge***. Basic LeSS is essentially an expanded version of a one-team Scrum. All teams are cross-functional, with few, if any, speciality teams because they are all focusing on implementing the same product. In short, in each sprint, all teams strive to create a common, shippable product, however, the sprint planning takes place in two distinct meetings, which is different from Scrum. The basic objective of LeSS is to simplify problems as much as possible by using "fewer roles, less management practices, and fewer organisational structures." Thus, LeSS can also mean "less".

Source: own processing according to: (Cluster, 2019, Shrivastava, A. et al., p. 1–4, 2021, agilest.org, 2019)

4.2 Transformation to Agile Methods

Agile transformation is the process of converting an entire organisation to a responsive, flexible strategy based on Agile principles. Implementing Agile methods does not just mean changing the process or the boxes on the forms. The initiator of the whole change is usually a pressing and seemingly unsolvable problem that is bothering an organisation so much that it is worth switching to Agile methods. Any change is hard and culture change is the hardest one. The change consists of changing the firm from a hierarchically managed firm to a problem-oriented firm. A properly Agile firm achieves what is known as a self-organised team. It means that there will be a change within teams and suddenly the work in question will not be based on the work of individuals directed from somewhere above, but on the cooperation of people in the team, on their ability to agree, decide and take responsibility for their decisions (Šochová, Kunce, 2019, p. 23).

Adapting to change is very challenging for many people, it means, that application of Agile may have many challenges and barriers for employees to adopting Agile. According to the annual State of Agile report 2022, three major challenges to an organisation's successful adoption of Agile are ***culture***, ***leadership***, and ***consistency***. In the survey about barriers to adopting Agile specifically on the business side of the organisation, the major obstacles to adopting Agile were a lack of leadership participation (42%), a lack of Agile understanding (40%), a general organisational aversion to change (40%) and not enough management support and/or sponsorship (39%). It can be said that insufficient leadership involvement, according to more than 2 in 5 respondents, is a barrier to the organisation's business side adopting Agile practices, while 2 out of 5 people cite a lack of management support and/or sponsorship as a hindrance. Company culture is another issue that came up, in addition to inadequate Agile leadership. Since Agile principles typically conflict

with corporate culture, studies reveal that 40% of people in their organisation are not content with Agile. The leading "cause of failed delivery with Agile" is identified as the corporate culture. In summary, the two key factors contributing to failed Agile delivery are company culture and a lack of management support. Furthermore, other barriers that may arise during the adaption of Agile in an organisation include a lack of wider buy-in around Agile, resistance to change, lack of open communication, poor collaboration and knowledge sharing, or pervasiveness of traditional development methods (Digital.ai, 2023, Santos, P., Carvalho, M., p.122–124, 2021).

On the contrary, one of the most common reasons for transition to Agile methods is ***flexibility***. Times have changed; nowadays, everybody wants everything right away. Most of the customers want to receive results in small chunks, ideally as soon as they are finished. Unfortunately, it is not possible in the actual process. Analytics have to describe the requirement, developers have to code it, testers have to test it, and that is not effortless. Another common reason is ***efficiency***. Studies show that multiple people working together is significantly more efficient than working alone. When it comes to frameworks, there are basically two options. The first one is to try pair programming, so having two people for all activities, including two software developers at one computer. Experience shows that this methodology is more efficient, better quality and faster than having each of them work independently. The second option is to build a collaborative team, as it is recommended by Scrum, where individual members work together on a single deliverable, help each other, and organise themselves. Obviously, it can take a while for the team to get used to each other, but it works great. ***Predictability*** is another important cause. Properly, projects should finish on time and without overtime before the end of the release, but this is not always the case. If this is not the case, Agile methods can help with this aspect. Agile methods introduce a completely different way of predictability. They estimate in relative units and involve the whole team in predicting. Breaking the project into small pieces is another important method to improve predictability. The so-called student syndrome is less pronounced on the short stretch and the effort the team invests is more stable, without fluctuations and stress at the end. What is also essential benefit of Agile is increased ***quality***. For quality, the company is targeting two areas. First, the customer is involved in the product. The company asks to a customer what he wants and why he wants it, and who will use the product and how. This shows results to the customer in parts and manages his expectations. It does not happen then, that the customer rejects the product as unusable and would insist on a complete change. And last but not least, work will be ***fun*** again. Even individual members will know what they are writing and why. They will understand the meaning of the product and understand the customer. In addition, working with others is also fun. A motivated team member is certainly more effective than a bored developer sitting alone at his computer. Indeed, before embarking on an Agile deployment, a company should get its expectations right, especially around flexibility, efficiency, predictability, quality and fun. Anyway, deploying Agile methods just because it's something new doesn't make sense. It's an arduous and thorny process, at the end of which a sufficiently large reward should await (Santos, P., Carvalho, M., p.124–125, 2021, Šochová, Kunce, 2019, p. 23).

In summary, this implies that there are many benefits that Agile can bring, such as increased flexibility, visibility, product quality, speed of delivery or customer satisfaction. As well as decreased

risks, faster process of work, better predictability and communication or collaboration within the team. In the 16th State of Agile study, more over half of the respondents stated that accelerating time to market is how they prioritize implementing Agile methods. Including the cost and revenue drivers, a key benefit that stands out from a host of others is the ability to move quickly while remaining predictable. Another advantage that is also frequently mentioned is risk reduction. In general, fast cycle times and the greatest financial and performance outcomes, each of which has been covered in more than ten publications, stand out among the business advantages. Benefits of products and processes, increased production, and better requirement management are more frequently highlighted. In particular, early Scaled Agile Framework adopters have noted a considerable improvement in terms of quality and productivity. As far as team benefits, the most mentioned advantages in publications are regular feedback and communication (Digital.ai, 2023, Santos, P., Carvalho, M., p.124–125, 2021, Laanti, M., 2014, p.9).

Understanding the organisation's motivation for switching to an Agile approach should be the first step in transforming to Agile, as each organisation may have a distinct one. The most obvious explanations are better delivery, higher product quality, and quicker market time. As critical success factors of Agile development, most Agile studies highlight top management support, team capability and training, team environment, Agile development techniques, customer involvement, project management process, organisational culture, communication, and openness to change. It is not even typical for an Agile approach to be naturally understood. Agile consultants and coaches should try to increase Agile adoption and keep their attention on ensuring that leadership and cross-functional colleagues comprehend what Agile is and how it will impact their job. They can do it by creating Agile training sessions and academies, various seminars, workshops, or even personal meetings. One of the other keys to Agile's success is top management support, which is significantly connected with the success of Agile projects, and from which flows confidence between the team's leader and members as well as among team members themselves. Next, any project's success depends on effective communication. Communication is a critical component of Agile development and essential to Agile initiatives' success. Effective communication and regular feedback from each iteration also often lead to better knowledge transfer. The Agile development team can produce improved estimates for each iteration thanks to short cycle periods and rapid feedback, especially after a few cycles of practice. Furthermore, along with addressing their crucial restrictions, the Agile strategy approach needs to address an organisational culture and management style. The managerial style, trust, acceptance of change, and extreme process emphasis are the most common cultural restrictions. Culture is one of the pillars on which the business strategy is built. Therefore, it is essential to be aware of the organisational structure in place as long as it reflects the chosen company strategies (Digital.ai, 2023, Popa et al., 2021, p.2–4, Santos, P., Carvalho, M., p.124–125, 2021, Koi-Akrofi, G., Koi-Akrofi, J., Matey, H., p. 25–44, 2019, Aldahmash, A., Gravell, A.M. and Howard, Y., 2017, p. 4–7).

4.3 Kanban

Kanban is a Japanese word which means “visual board” or a “sign”. It is a well-known management methodology built on the Agile development framework that helps organisations manage and improve work systems. Kanban is known as well-liked lean workflow management technique for establishing, managing, and enhancing services delivering knowledge work. Thus, Kanban helps companies with visualizing work, maximising efficiency, and improving continuously. Work is represented on Kanban boards, allowing companies to optimise the delivery of work across multiple teams and manage even the most complicated projects in a single environment (Kanbanize, 2023, Zayat, Senvar, 2020, Ahmad et al., 2018).

Since the 1950s, the term "Kanban" has been used to refer to the process definition. One of the biggest automakers in the world, the Toyota Production method, created and used Kanban first as a part of Lean manufacturing, a scheduling method for just-in-time manufacturing. The Kanban method is a "pull system," which implies that rather than the conventional "push system," where things are created and pushed to the market, the company's production is dependent on consumer demand. In general, lean manufacturing, or simply Lean, was established by Toyota's distinctive production strategy. The core idea of lean is to eliminate waste without sacrificing productivity and, as the goal, to increase customer value without increasing prices. Nevertheless, the "Kanban Method," which was first defined in 2007, is what is known and connected with the capitalised term "Kanban" (Kanbanize, 2023, Ahmad et al., 2018, Lei et al., 2017).

Agilealliance (2022) describes the Kanban method as a tool for designing, managing, and enhancing knowledge work-flow systems. The method also enables to organisations to begin with an existing workflow and make evolutionary changes. Companies can do this by visualizing their flow of work, reducing work in progress (WIP) and stop starting and start finishing. A general term for systems using the Kanban method is flow in the sense that workflows through the system continuously, rather than being organised into individual time boxes (Agilealliance, 2022).

One of the advantages of Kanban style approach is that it is easy to implement. It can be the biggest benefit for a lot of businesses. Consequently, there is no need for the whole team to attend a training course or to learn difficult new tasks. The original concept behind the Kanban method was to use sticky notes on a whiteboard to display activities' progress in a very visual manner. These days, a digital Kanban board is more typical because this method no longer serves the needs of the majority of modern businesses. Nevertheless, the straightforward method of tracking work is still in use. Another benefit is that Kanban can work in many different industries. The effectiveness of the Kanban method of task management has been demonstrated across a wide range of sectors. It is frequently used in software development and may be also effective in industries like finance, marketing, and healthcare. It works especially effectively with businesses that implement the Agile methodology of project management. Moreover, Kanban also encourages collaboration among team members, which is one of the major challenges that businesses need to overcome in the contemporary world. Everyone can see how their own tasks fit into the larger workflow and who is awaiting the output in order to complete their own jobs. In short, the key benefits of Kanban include

reducing customer defects, reducing lead times required to deliver the product, enhancing product quality, emphasising team member collaboration and communication, and achieving consistent product delivery (Ahmad et al., 2018, Kanbanchi, 2022).

On the contrary, the disadvantage of Kanban might be concerned that it might be less useful in a dynamic setting. In a work setting that is largely steady with little changes that complicate things, using a Kanban board is excellent. However, it can be less effective in a fast-paced, dynamic environment where things change from day to day and where initiatives can take on a different form every day. Another drawback to keep in mind before beginning is that a Kanban board does not show how long it will take to complete each activity. There are no deadlines specified for when they must be finished; they simply advance across the board from one column to the next one. In actuality, Gantt chart should be considered if you wish to assign timeframes to a number of tasks (Ahmad et al., 2018, Kanbanchi, 2022).

Many people take Kanban directly as an Agile methodology, which, as was mentioned, uses visual cues to manage the flow of work. In some publications, Kanban is identified as one of the best-known Agile methodologies, along with Scrum, which is a framework of doing project by dividing tasks into manageable chunks called sprints. In a framework study for Agile software development via Scrum and Kanban, results of a focus group survey demonstrate the adaptability of both methodologies in achieving Agile goals, while Kanban is more effective in continuous flow situations with a consistent attitude toward system improvement, Scrum emphasises the corporation of the customer and development teams with a focus on specific skills (Zayat, Senvar, 2020).

Comparing Kanban and Scrum methods, the research of Lei et al. (2017) aimed to compare the two methodologies. The impact of Scrum and Kanban on software development initiatives was statistically analysed. The authors' analysis was based on a survey in which 35 participants—60% of whom used Scrum and 40% of whom used Kanban—were asked to respond to a series of 14 questions covering six key areas: risk, quality, resources, schedule, scoop, and budget. Their findings demonstrated that Kanban outperformed Scrum in terms of scheduling, the length of time required for projects, product quality, resource management, and risk management. Scrum, on the other hand, was better at managing the budget and the investment's return. Nowadays, Scrum and Kanban methodologies are the most widely used Agile methodologies by system development organisations all over the world (Lei et al., 2017).

4.3.1 Kanban Values

Kanban can be used effectively in various knowledge work environments, especially when work items are irregularly generated or when it's necessary to start working on tasks as soon as they're ready, without waiting for other items to be completed. Teams that implement Kanban with the aim of enhancing their services uphold certain values.

Transparency in meaning of open communication and the use of clear and straightforward language that enhance the flow of business value. It combines the three concepts of process visualisation, the introduction of clear rules, and the development of feedback loops. Teams benefit from Kanban's openness on a number of levels, including work visibility, status updates, process understanding, awareness of the decision-making process and its potential for change, as well as access for customers to see how the process is carried out. These factors make Kanban the ideal method for developing flexible, adaptable systems and determining the most effective development approaches (Burrows, Kanban Tool, 2022, Kanbanize, 2023).

To be effective, several factors, perspectives, and capacities must be balanced, thus, next Kanban's value is ***balance***. An organisation can reduce the cycle time and delivery rate by successfully implementing WIP limitations, which also includes routinely testing for relevance. The team and the client are kept satisfied by preventing the beginning of new work, which helps to maintain a balanced flow. Additionally, the team is able to make balanced decisions regarding executing the actual work and taking actions to enhance the process itself, allowing for variation in what they do, as a result of better management of what the team is working on (Agile Alliance, 2023, Burrows, Kanban Tool, 2022, Kanbanize, 2023).

Value that relates specifically to the "value" that the customer perceives is ***costumer focus***. Establishing a policy that will enable activities to be completed on time is not sufficient; rather, an organisation must complete tasks to the full satisfaction of the customer. It is intended to fulfil a specific need for the customer. It has been a waste of time and resources if the consumer does not receive benefit from it. The goal of Kanban systems is to optimize the flow of value to customers who are external from the system but may be internal or external to the company in which the system is included (Burrows, Kanban Tool, 2022, Kanbanize, 2023).

Kanban's known value is also ***flow*** since work is an ongoing or serial flow of value. Flow gives the impression that a process is gently onging and somewhat predictable, allowing for the management of any obstacles and problems that may arise (Burrows, Kanban Tool, 2022, Kanbanize, 2023).

Important value is ***leadership***. At all organisational levels, leadership, which is the ability to motivate others by examples, words, and reflection, is required in order to achieve continuous development and deliver value. A successful leader encourages self-organisation and the capacity for change in each team member. Additionally, since all of these circumstances foster successful change, it is advocated that the leader both challenges and is prepared to be challenged (Agile Alliance, 2023, Kanbanize, 2023).

To advance and progress, it is essential for both individual and organisations to be aware of the starting point and it's about ***understanding***. Understanding refers to having a clear understanding of how a process operated prior to a company deciding to impose Kanban on it. It also applies to learning about and comprehending the organisation's overall operations and change

management philosophy. The likelihood that it will succeed increases with the depth of understanding of what is going to change (Burrows, Kanban Tool, 2022, Kanbanize, 2023).

Moreover, ***agreement*** is also essential as everyone involved in a system is dedicated to its improvement and agrees to work together to achieve its objectives while respecting and tolerating difference of viewpoint and approach. This briefly discusses situations in which changes are imposed from without and for which there is no internal consensus or understanding. Additionally, it frequently happens that people acknowledge the existence of a workflow issue but refuse to agree to implement the necessary corrections (Burrows, Kanban Tool, 2022, Kanbanize, 2023).

Another important Kanban value is ***collaboration*** – Kanban was developed to facilitate better teamwork. A team is expected to work together and go outside of their inner team to collaborate develop answers and make plans for organisational and process improvements, in accordance with the prior principles of agreement, respect, and customer focus (Burrows, Kanban Tool, 2022, Kanbanize, 2023).

And lastly***, respect*** – value, understand, and be considerate of people. As a pillar of Lean, it calls for recognising the roles and responsibilities that already exist inside the organisation that is undergoing transformation. By showing respect for how the team is currently working, there is a better chance of gaining the team's willingness to apply the new approach (Agile Alliance, 2023, Burrows, Kanban Tool, 2022, Kanbanize, 2023).

Figure 10 Kanban Values

Source: own processing according to: (Kanbantool, 2022, Kanbanize, 2023)

4.4 Scrum

Scrum is the most well-known Agile methodology, which is often mistakenly used as a synonym for Agile. Scrum is described as a methodology based on the Agile principles that enables people to handle complicated issues while still producing high-quality products at a high rate of productivity. Scrum was first introduced in 1995 by Jeff and Ken Sutherland at the object-oriented programming, systems, languages, and applications (OOPSLA) conference in Austin, Texas. The methodology was then illustrated in a paper written by Ken Sutherland. The entire concept was based on another article in which the term Scrum was used because of its connection to the sport of rugby, where it refers to a team structure in which each player has a specified function and swift strategy adoption is encouraged, to emphasise the significance of collaboration between team members for successful projects outcomes. Sutherland's work introduced a novel concept that can be used to great benefit in challenging situations. The goal is to create structured teams with a set of goals rather than tasks. Afterwards, the first Scrum book was published. Once the Scrum Alliance was established in 2002, certified Scrum-Master training were offered all around the world (Zayat, Senvar, 2020, Schwaber, Sutherland, 2017).

Empiricism, often known as empirical process control theory, is the basis of Scrum. According to empiricism, knowledge is derived from experience and decisions that are based on what is known. Every application of empirical process control is supported by the three pillars of transparency, inspection, and adaptability. Scrum Teams, along with the roles, events, artifacts, and rules they relate to make up the Scrum Framework. Each element of the framework has a distinct function and is necessary for Scrum to function and be used. Small, highly adaptable teams of people make up the core of Scrum. Scrum Teams produce things incrementally and iteratively, maximising the chance for feedback. Scrum Framework is used for creating, delivering, and maintaining complex products. It is a framework that enables individuals to manage complicated adaptive challenges while effectively and innovatively producing goods of the greatest quality. It is characterized by brief project cycles, or "sprints," during which a completely planned, developed, constructed, tested, evaluated, and launched a usable product. In a nutshell, Scrum is a lightweight, easily understood, and challenging to master methodology (Mihajlovic-Milicevic et al., 2019, Schwaber, Sutherland, 2017).

Regarding Agile team roles, which are described individually in the previous chapter, a Product Owner, a Scrum Master, and the development team are part of each Scrum team. Scrum teams are cross-functional and self-organised, which means that instead of being directed by others, they decide how to perform. Teams are made to be adaptable, highly creative, and highly productive. Customers have the chance to provide feedback at each iteration because products are delivered iteratively (Zayat, Senvar, 2020).

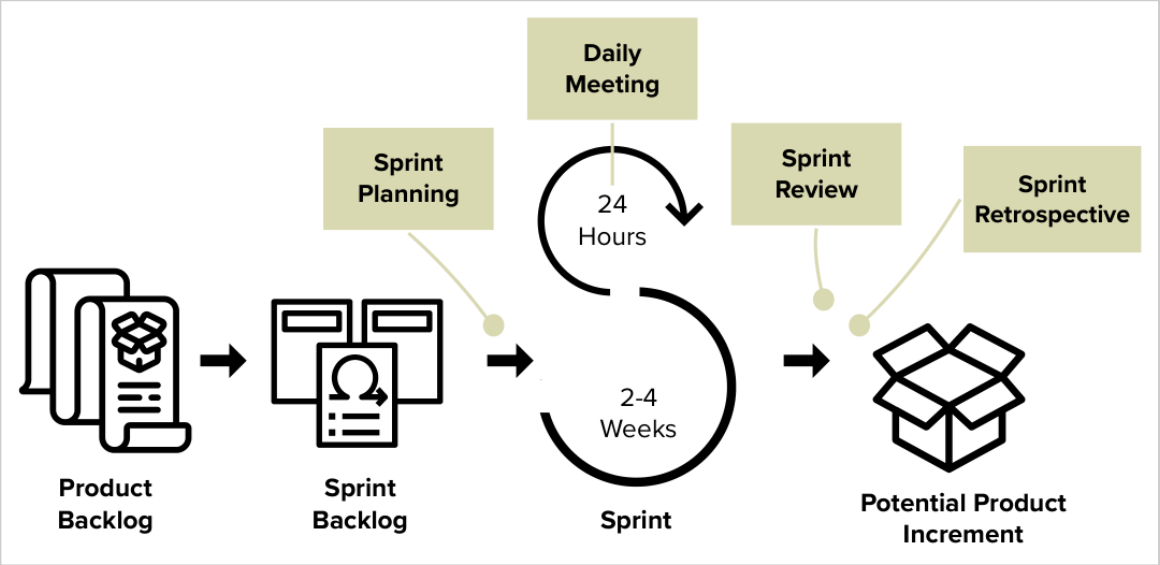
In comparison with Kanban methodology, in terms of similarity, the Agile 12 principals as outlined in the Agile Manifesto form the foundation of both methods. Also, the Kanban method and the Scrum framework use user stories to divide projects into small manageable tasks, presented by cards in Kanban system, and by the mean of user stories in Scrum. Moreover, both

operate as pull systems that take orders from customers, and both place a great importance on process optimisation, ongoing progress, and the quality of the work. The visual representation of cards on the board in Kanban and the daily face-to-face meetings or, once more, the use of the board in Scrum both promote transparency. As far as development teams, Scrum teams as well as Kanban teams are efficiently managed and self-organised. A possible Agile approach is also ***Scrumban***, which is the combination of set of components from Scrum and Kanban. Although Scrumban does not include sprints, it contains methods that are most closely associated with Scrum development, such as daily stand-up meetings, user stories, and aspects of self-organised teams. Scrumban was originally designed as a way to transition from Scrum to Kanban (Zayat, Senvar, 2020, Yordanova, Toshkov, 2019, Kumar et al., 2019).

On the contrary, when it comes to the differences between Scrum and Kanban, Scrum promotes user input and involvement in the development process more than Kanban. Even though using Kanban as an Agile methodology necessitates scheduling in-person meetings with clients, these encounters are not as frequent as those in Scrum. Nevertheless, Kanban is a more flexible system than Scrum. There is more flexibility in the boards, operations, and delivery schedule. According to Kumar et al. (2019), Kanban can be exposed to frequent changes in the specifications, outperforming Scrum at this time, which can only handle medium level modifications. Further, every team in Scrum has a clear job to play, the project owner determines what has to be done, the Scrum Master manages the time progress, and development teams complete the task; however, in Kanban, a project manager may be present to oversee the entire process. Scrum teams are cross-functional; they need to communicate with one another, but each member has his own set of tools. While using Kanban, every team uses the same board, and each team focuses on a specific task (Zayat, Senvar, 2020, Kumar et al., 2019, Yordanova, Toshkov, 2019).

In general, Scrum is also considered as one of the most popular project management methodologies that relies on incremental and iterative progress. The first stage in the Scrum paradigm is the generation of the product backlog. After that, the construction of the sprint backlog and sprint planning are used to process the second stage. Each sprint begins with a sprint planning session, during which the Scrum team convenes for a lengthy meeting, which could last four hours, to thoroughly plan the sprint. The third stage involves planning meetings and sprints. The project will then be developed in sprints, with each sprint typically lasting two to four weeks. Every day, a 15-minute meeting is often scheduled with three primary topics: what was accomplished yesterday, what will be accomplished today, and what pressing issues need to be resolved. After each sprint, a performance test is conducted in the fourth step. The fifth stage completes the retrospective and plans the following sprint. Stakeholders join a review meeting at this point to evaluate the sprint's teamwork. Besides, customer requirements are displayed in the Scrum backlog, and daily burn-down charts illustrate the work still to be completed (Kumar et al., 2019, Tavares et al., 2019).

Figure 11 Scrum Life Cycle



Source: own processing according to: (Tavares et al., 2019)

Everything begins with organising the product backlog, which is done with the Scrum Master's oversight. After that, the sprint planning and actual sprint begin, producing a deliverable product that is prepared for the customer. The same cycle then repeatedly repeats after that. The Scrum framework, which contains these steps, is illustrated in picture above *(Figure 11)*. The individual Scrum Events and Scrum Artifacts are described in more detail in the following subchapters (Tavares et al., 2019).

4.4.1 Scrum Events

To control and adapt the process, the Scrum method prescribes four formal activities, which are ***Sprint Planning***, ***Daily Scrum (StandUp)***, ***Sprint Review*** and ***Sprint Retrospective***:

The work to be done within a sprint is planned in a ***Sprint Planning*** meeting, where the planning is done by the entire Scrum team. The meeting starts with the Product Owner presenting the sprint goal to the team, but this goal should not be completely new to the team as the team should already have knowledge of the product backlog. The Scrum Master moderates the meeting, and especially with start-up teams, needs to be proactive in helping the team understand what they are committing to in order to make their commitment realistic. During the meeting, the Development team will select the items from the product backlog that they want to commit to completing by the end of the sprint. The Product Owner only determines the priority, but the amount of work done is determined by the Development Team. Breaking it down into individual tasks is also part of sprint planning, although some teams leave this part for the first day of the sprint. Tasks should be roughly one day, two days at most. At the end of the meeting, each team member should verbally agree to the commitment of the sprint, and it is also a good idea to make sure that each team member has at least one task to start working on. When the Development

team comes to a commitment – they will select a number of ***User Stories*** for the upcoming sprint – they should still have the opportunity to discuss their decision without the Scrum Master and Product Owner for some time before confirming or changing their commitment. Also related to sprint planning is ***Backlog Grooming***, which is a meeting to better understand the content of the product backlog, where the Product Owner goes through the various User Stories, explains their importance and the team evaluates the new User Stories (Radigan, 2021, Šochová, Kunce, p.119, 2019, Schwaber, Sutherland, 2017).

Daily Scrum (Stand-up) is a regular daily meeting that usually takes place in the morning at the Scrum board. The stand-up always takes place at the same time in the same place, and each member of the Development team talks about what they have done during the previous day, what work they will be doing on the current day and any obstacles that are preventing them from working. The purpose of this meeting is to review the previous day's progress, plan the current day's work, and look for ways to make the most progress that day through collaboration. The entire meeting should last no more than 15 minutes. If the discussion or resolution of a problem takes more than a minute, a separate meeting should be scheduled for that issue, but the resolution should not occur during the Stand-up. The Scrum Master is only in the role of moderator or facilitator, and if someone outside the Scrum team attends the meeting, it is only in the role of an observer and does not interfere with the proceedings. Properly executed Stand-up meetings improve communication, eliminate extra meetings, identify obstacles, promote quick decision making and improve the Development team's knowledge level (Šochová, Kunce, p. 107, 2019, Schwaber, Sutherland, 2017).

Sprint Review is held after the end of the current sprint and before the start of the new sprint. During this meeting the Development team demonstrates the result of their work. The meeting is attended by the Scrum Master, the Product Owner, the Development team, but also, for example, by customers or developers from other projects. In this context, a customer is anyone who has any interest in the product. The concerned customers are usually not interested in individual User Stories, but in the overall increment of the product, so it is advisable to involve the customers as much as possible and let them try out the new functionality. Sprint review is an informal meeting to elicit feedback and encourage collaboration. The actual presentation should be done directly by the Development team, as team members know best how the product works and what they have to show the customer. Only fully finished User Stories belong on sprint review (the term "***done-done***" is used), these are User Stories that meet the Definition of Done and are already accepted by the Product Owner during the sprint. It is also important that the presented product is reviewed in terms of the sprint goal so that it can be confirmed that the team has achieved the goal. Furthermore, the whole group discusses ideas at the meeting that can help define content for the next sprint. The development team also talks about what went well during the sprint, as well as the problems they had to solve, which is crucial for next sprints. However, the Product Owner shouldn't learn about these issues until the sprint review, the team should keep them informed on an ongoing basis. During the sprint review, the Product Owner should receive feedback from the stakeholders and development team. The Product Owner usually talks about the status of the product backlog and the completion dates based on progress to date. Additionally, to

facilitate the meeting, maintain the timeframe, and inform all parties of the purpose and goals of the meeting is the Scrum Master's job. There should be some ground rules for a conversation to keep the meeting going well (Zayat, Senvar, 2020, Šochová, Kunce, p. 125, 2019, Schwaber, Sutherland, 2017).

After the sprint review, but before sprint planning, there should be a review of the completed sprint – the ***Sprint Retrospective***. At this meeting, the Scrum Master should help the team reflect on the lessons learned, as the main goal of this meeting is to improve through the experience gained. The sprint retrospective provides a formal opportunity to focus on review and adaptation at the level of people, relationships, processes, and tools. The sprint retrospective can help motivate Development Team members by providing an opportunity to speak up, share ideas, and be heard at the same time. It is claimed that many Agile leaders don't understand the true value of this meeting at first and don't hold these meetings at all because they mistakenly consider them a waste of time. These meetings are a great opportunity to appreciate the contribution of all team members as well as to reveal the time and financial risks. During the retrospective, the Scrum team also plans how to improve the quality of the product by modifying the ***Definition of Done (DoD)***, which is a collection of requirements that must be met for a project to be considered “done”. According to a study by Andrea et al. (2019), the sprint retrospective is the proof that Scrum is an adaptable framework that allows the implementation of the necessary changes. The meeting can have different formats, anyway the most common way of collecting data is the so-called "round robin", where each team member answers three questions in turn - what they liked and want to continue, what they did not like and would like to change, and what they would like to introduce new. In the next phase, the Scrum Master summarizes everything the team members have said and lets the team discuss and brainstorm, ask for clarification if necessary, and then the team chooses the specific actions they want to implement in the next sprint. The Scrum Master's job is again to ensure the meeting is held, that the length of the meeting is met, and that the team understands the purpose of the meeting (Andrei et al., 2019, Schwaber, Sutherland, 2017).

4.4.2 Scrum Artifacts

Scrum's artifacts serve as representations of work or value and offer chances for inspection and modification. Scrum-defined artifacts are specifically designed to maximise the transparency of key information so that everyone understands the artifact in the same way. These practices and artifacts primarily include ***Product Backlog***, ***Sprint Backlog***, ***Sprint,*** and ***Product Increment***:

Product Backlog is an orderly list of everything that is known to be required for the product and the only source of requests for any changes. The product backlog is the responsibility of the Product Owner. The Product Owner makes a list of the elements from user stories and prioritizes them correctly. This is accomplished by ranking the features or components that must be on the product in order of importance, then those that are less crucial, and ultimately those that are not feasible within the given timeframe. This will guarantee the project's aspects' clarity and is done based on the items' impact, risk, and contribution to the project's learning process for the final product. The product backlog is never complete, it evolves and changes as the product and

environment evolves. The ***Product Backlog Item*** is an independent functionality that adds value to the customer and therefore can be feedback on. Most often, these items are described in the form of a User Story, that are described from the customer's perspective, not from the perspective of the technology or system. At any point in time, the remaining work to achieve the goal can also be calculated, and the Development team is responsible for estimating the difficulty of the product backlog items. However, the product backlog usually does not include individual tasks because it is an unnecessarily high level of detail that belongs on the ***Scrum board***. The product backlog should be prioritized, with the most important ***User Stories*** at the top, defined so that the team can accept them into the sprint Backlog and implement them within the sprint. According to the authors, the work for 2 to 3 sprints ahead should still be prepared in this detail, with a lower level of detail the work for the next 5 to 10 Sprints should be defined, and the remainder can be left at the level of ***Epics*** – large items that still need to be split into User Stories (Zayat, Senvar, 2020, Mihajlovic-Milicevic et al., 2019, Šochová, Kunce, p. 67–75, 2019, Schwaber, Sutherland, 2017).

Sprint Backlog is a small part of product backlog that contains functionality planned for one sprint. The creation of a sprint Backlog occurs during sprint planning, where the Product Owner presents a list of prioritised items from the product backlog and the Development team analyses the list. It determines how much work it can do and prepares an implementation plan. With the sprint backlog, all the work that the team has identified as needed to meet the sprint goal is clearly visible and only development team can change the contents of the sprint backlog during the sprint. Although Scrum does not require the team to create individual tasks most teams do that. According to some sources, breaking it down into individual tasks increases visibility into risks and the current state of work. However, these tasks are not part of the sprint backlog and are just internal practices of the team. The development team also monitors the remaining work in the sprint backlog in daily meetings – Daily Scrum – to check the chances of achieving the sprint goal (Šochová, Kunce, p.75, 2019, Mastropasqua, 2018, Schwaber & Sutherland, 2017).

In the stage of ***Sprint***, the developing team is working on the user stories that were chosen for the product's subsequent shipment. During the planning meeting, these stories are picked from the product backlog based on priority and the expected amount of work required, which should fit inside one sprint. A daily meeting is held to ensure that the sprint is proceeding according to schedule. Except for specific blocks of outstanding items, the sprint backlog cannot be changed while the sprint is in progress. The features that are added during a sprint are finished, tested, and prepared to be delivered as the new product increment. A sprint is seen of as a project with a specific goal and scope. Sprints often last two to four weeks, but rarely longer than one month. As the team learns more about the project, this time frame may be altered under the direction of the Product Owner. When a sprint lasts a long time, the product backlog may alter, increasing the project's risks, expenses, and complexity. A sprint can be finished after the work is finished and the project is prepared for delivery. At this point, the client or the shareholders can run certain tests. Automation can be useful in this situation because it can help with the time and money problems. Immediately following the conclusion of the preceding sprint, a new sprint begins. The Sprint Planning, Daily Scrums, Development Work, Sprint Review, and Sprint Retrospective are all parts of a sprint (Mihajlovic-Milicevic et al., 2019, Schwaber, Sutherland, 2017).

Product Increment is the total of all backlog items completed during the sprint as well as the previous successful sprints. Each sprint should end with a working product that can enter the production phase and go live. In summary, product increment is what gets produced at the conclusion of a development cycle or timebox (Andrei et al., 2019, Mastropasqua, 2018).

To conclude, the same Scrum cycle, containing the Scrum Events and Artifacts, is continually repeated, beginning with the items at the top of the product backlog, going through the same phases, and improving the process by trying out new ideas and honing work to a high level of performance. Additionally, only properly qualified teams should use Scrum. Scrum applications frequently underestimate schedules and budget, which is a concern. The team velocity chart and the burn out chart can be used to evaluate the performance of Scrum. These two charts were presented by Dixit and Bhushan, who also created several measures to gauge the efficacy of sprint planning and software problems in Scrum applications (Emelyanova et al., 2020, Dixit, Bhushan, 2019).

MSD CASE STUDY

- PRACTICAL PART

5 Company Introduction

Merck & Co., Inc. is an American global pharmaceutical corporation with its headquarters in Rahway, New Jersey, named after the Merck Group, which was established in Germany in 1668 and of which it was formerly the American arm. In the United States and Canada, the corporation is known as Merck Sharp & Dohme, while outside of North America, it is known as MSD. The official name is Merck & Co., Inc. It is one of the biggest pharmaceutical corporations in the world, regularly placing in the top five globally in terms of sales, with over 68,000 employees (as of Dec. 31, 2021). The business provides cutting-edge health solutions through its biologic therapies, prescription medications, vaccinations, and animal health products. In order to provide innovative health solutions, the organisation operates in more than 140 countries. Since 1992, MSD has been conducting business in the Czech Republic. With more than 70 various nationalities, MSD Czech Republic employs over 1,600 people, 42 per cent of whom are women. The Top Employers Institute has recognised MSD Czech Republic as one of Europe's Top Employers for the years 2021, 2022, and 2023 (msd.com, 2022, msd.cz, 2023).

Figure 12 MERCK logo

Source: logos-download.com, 2016

Figure 13 MSD logo

Source: logos-download.com, 2016

In the pictures above *(Figure 12 & Figure 13)* is illustrated the current Merck & Co. logo, created in 1965 by Chermayeff & Geismar. In the United States and Canada, the "MERCK" lettering is used (left); elsewhere, the "MSD" writing is used (right).

The aim of the business is to apply cutting-edge science to save and enhance lives all over the world. Through creating significant drugs and vaccines, the firm has given humanity hope for more than 130 years. The company aspires to be the world's top biopharmaceutical research-intensive company, and it is now leading research to produce cutting-edge health solutions that improve illness prevention and treatment for humans and other animals around the globe. To ensure a secure, sustainable, and healthy future for all individuals and communities, MSD promotes a diverse and inclusive global workforce and conducts ethical business every day. The business is ranked 71st on the 2022 Fortune 500 and 87th on the 2022 Forbes Global 2000, each based on 2021 revenues (msd.com, 2022).

In conclusion, the company creates and manufactures pharmaceuticals, vaccines, biologic treatments, and animal health products. It has a number of top-selling medications or products, each with 2020 revenue projections, including as cancer immunotherapy, diabetes treatment, and vaccines for HPV and chickenpox (msd.com, 2022).

5.1 Company History

Since 1891, the company's researchers have contributed to the development of numerous medical advancements, including the first statins to treat excessive cholesterol as well as the first measles vaccine and the discovery of vitamin B1. Scientists at MSD/Merck have also contributed to the development of numerous items, including as vaccines and antibiotics, to enhance animal health. The organisation is now focused on creating significant scientific discoveries in the fields of hepatitis C, HIV, diabetes, and immuno-oncology as a leading research-intensive biopharmaceutical enterprise (msd.com, 2022).

For more than 130 years, the company has been led by the belief that effective drugs and vaccines can alter the world. On January 1, ***1891***, Merck & Co., Inc. was established in Rahway, New Jersey, USA. In order to distribute quality chemicals throughout New York City and the surrounding areas, then 23-year-old George Merck founded the corporation in the United States. The Merck Manual was initially produced by the business in the United States in ***1899*** (now known as The MSD Manual outside the U.S. and Canada). Arsenic for impotence, bloodletting for acute bronchitis, and almond bread for diabetes were all prescribed treatments in the original guidebook. The Manual later developed into one of the most popular medical dictionaries in the USA. The first MRL Research Laboratory was established in ***1933*** in Rahway, New Jersey as the company's first step into pharmacological research. Next, in ***1957***, MSD donated $500,000 to launch the company's Foundation, a non-profit organisation devoted to humanitarian contributions. The Foundation has given non-profit organisations hundreds of millions of dollars to date. Further, the first pneumococcal vaccine was authorized in ***1977***. On April 29, ***1992***, Merck Sharp & Dohme IDEA (MSD) opened a branch in the Czech Republic. MSD quickly rose to prominence and is now among the biggest pharmaceutical firms in the Czech Republic. Biological treatment, oral antidiabetics, cardiology, the treatment of infectious diseases, women's health, and vaccines are just a few of the areas in which MSD has established a strong position over time. Then, in 1992, the first statin medicine to lower levels of "bad" LDL cholesterol was released by MSD. Furthermore, MSD launched the first cancer vaccination to the Czech Republic in ***2006***, called Gardasil (formerly Silgard), which guards against human papillomavirus (HPV) cancer. Gardasil 9, an advanced HPV vaccine that offers even more protection against HPV virus, was released by MSD in 2015. Moreover, in 2006, Cured MSD introduced a ground-breaking medication for people with type 2 diabetes based on incretins to help lower and maintain healthy blood sugar levels. One of the numerous prestigious Prix Galien Awards was given to MSD in ***2007*** for this drug as the best pharmaceutical the business had ever produced for medical use. Beginning in ***2011***, the global MSD for Mothers campaign. In 2010, one woman died during childbirth or pregnancy every two minutes. Several of these deaths could have been avoided. In response, the company introduced MSD for Mothers, a global project with partners to enhance the health and wellbeing of mothers prior to, during, and following childbirth. The campaign has reached more than nine million women worldwide in 48 countries as of 2019. The year ***2016*** saw significant advancements in immuno-oncology. MSD has unveiled a new immunotherapy therapy that can be used to treat different malignancies (msd.com, 2022, msd.cz, 2023).

One day, during a speech at the Medical College of Virginia in Richmond, George W. Merck made a famous claim about how the medical and pharmaceutical industries could succeed that would come to define the company: "*We try to remember that medicine is for the patient. We try never to forget that medicine is for the people. It is not for the profits. The profits follow, and if we have remembered that, they have never failed to appear."* (1950, December 1, George W. Merck). Leaders and staff of MSD still adhere to this idea. It means that the organisation's legacy of developing drugs and vaccines is still going strong today. MSD adapts its business not just for the upcoming quarter, but also for the upcoming quarter century (msd.com, 2022).

5.2 Company Culture and Values

The ***vision*** of MSD is to improve people's lives all over the world with its innovative medicines, vaccines, and animal health solutions. MSD is committed to being the leading, research-intensive biopharmaceutical firm and its staff are committed to offering cutting-edge technologies and solutions for both the present and the future. The company's ***mission*** is to find, create, and offer innovative products and services that enhance and save lives all over the world. MSD does ethical business every day to support a safe, healthy, and sustainable future for everyone. Everything revolves around preserving and enhancing life. Whatever the position or designation, MSD rises to the challenge in the quest for improved health outcomes (msd.com, 2022).

The following are the MSD values: ***Patients first*** – offering high-quality goods and services, enhancing the health and wellbeing of people and animals everywhere, and increasing access to the company's medications and vaccines; ***Respect for people*** – fostering an atmosphere of tolerance, accountability, and inclusivity while also acknowledging the needs of MSD employees and their families; ***Ethics and Integrity*** – adherence to the highest ethical and moral standards, as well as accountability to all stakeholders. ***Innovation and Scientific Excellence*** – identifying and addressing the most pressing requirements of patients and clients through ongoing innovation across the whole MSD business (msd.com, 2022).

The values of MSD serve as the foundation of the company's identity and direct all decisions and actions. Everything MSD does, both inside and outside of the lab, is predicated on a profound enthusiasm for life. MSD has gathered for this reason, with a single purpose – to use leading-edge science to enhance and save lives all throughout the world (msd.com, 2022).

5.3 MSD IT

Through digital innovation, MSD is transforming healthcare. The company develops multi-platform apps and mathematical models for sales, research, business analysis, and the development of medicine, leveraging its experience in software engineering, networking, and digital marketing to serve the entire organisation. ***Software Engineering*** is where innovation begins. MSD assists its business in bringing the most efficient therapies to market by developing the websites and mobile applications that support the company's digital transformation. Next MSD

expertise is ***User Experience*** that defines and jointly develops intuitive, useable digital goods with product teams to guarantee that all the tools and solutions MSD offers are user-friendly and enjoyable to use. Next part of MSD expertise is ***Agile Development***, which entails mentoring and inspiring MSD product teams by promoting experimentation and setting a positive example. The MSD teams' ultimate objective is to produce demonstrable value. ***Data Science***, a department within MSD IT, uses cutting-edge data management, visualization, and mathematical modeling approaches to address problems and accelerate the delivery of novel medicines to patients while enhancing financial performance. Further, ***Research & Lab IT***. In addition to working with other IT professionals, scientists, and research teams in Prague and elsewhere in the world, Lab IT supports the technologies utilised in the MSD research and development process. When it comes to ***Manufacturing IT***, MSD supports manufacturing facilities all over the world for the most reputable supplier of drugs, vaccines, and biologics for both human and veterinary healthcare products. Another component of MSD IT is the ***Project Management Office*** which adopts a structured approach to managing the MSD portfolio to support teams in delivering outcomes by defining value measurements and coordinating all initial investments with the strategic goals of the company. The ***Infrastructure & Operations*** teams are the next section of MSD IT which mission is to put together and run a 24/7 IT Services model that is sustainable. Teams plan, manage, and constantly enhance IT services around the globe by utilising data and analytics to drive more effective operations. And last but not least***, IT Risk Management & Security*** which is responsible for leading teams in the joint responsibility of preserving the company's strategic and scientific information. Teams are actively devoted to preserving compliance at all times while assuring the availability, integrity, and confidentiality of the company's data (msd.cz, 2023).

6 Case Study

The second part of this thesis will focus on the empirical aspect of the study, which pertains to the case study of the company previously presented in the preceding chapter, MSD. The chapter will begin with a detailed description and justification of the research methodology employed, followed by a description of the preparatory phase of the case study, where research questions were formulated, and the implementation phase. In the end, the analysis and evaluation of the results obtained through the questionnaire survey will be presented. Additionally, the findings of the case study will be briefly compared with another public study. Finally, this chapter will conclude with a set of recommendations and suggestions based on the data collected.

The data collection for the case study was conducted through a questionnaire survey administered within the MSD company. Supplementary data were obtained through interviews with key stakeholders. It is important to note that all sensitive company information, including but not limited to the names of projects, products, respondents, Agile coaches, employees, and Agile teams, will be kept confidential in accordance with internal corporate policies.

By utilising a combination of primary and secondary data sources, this study aims to provide a comprehensive and in-depth analysis of the MSD case study, which can serve as a valuable reference for practitioners and researchers in the field of Agile Development.

6.1 Research Methodology

The research methodology employed in the practical part of this thesis is a mixed research design, combining both qualitative and quantitative methods. The case study is practice-oriented research, which is a qualitative research methodology that examines how MSD IT staff members implement Agile approaches within the company. Specifically, this case study will investigate the ways in which MSD IT staff members utilise Agile approaches in their daily work. The primary method of data collection for the case study is a quantitative questionnaire survey. Additionally, some data and extra insight will be gathered through interviews with the key stakeholders, such as an Agile manager and members of the Agile Development team.

The decision to focus on this particular case study topic was motivated by personal experience gained through part-time employment at MSD, where I have been working for over a year in the Agile team within MSD IT as an intern. Furthermore, my personal and professional interest in Agile coaching and related topics also contributed to the selection of this topic. At the same time, this theme is also relevant and essential for MSD itself since the company has only recently begun to adopt Agile approaches, and no previous studies or surveys have been conducted regarding the use of Agile methods at MSD. Agile techniques experienced significant growth within the MSD software development sector in 2018–2019 with the establishment of an internal software development team (Global Software Development, GSW). Subsequently was created the first independent team of Scrum Masters and Agile coaches, and secondly, a team of product managers.

The primary aim of the MSD case study is to investigate the utilisation of Agile practices and their efficacy for MSD IT members. The study also seeks to identify the frequently adopted Agile frameworks/methods and Agility-supporting tools used by MSD IT employees. Moreover, the investigation aims to uncover the key factors that MSD IT staff consider critical to the successful implementation of an Agile approach, as well as the significant barriers to the adoption of Agile practices in MSD IT teams. Other sub-objectives include examining how respondents rate the benefits of the Agile approach, the impact of Agile development practices on the outcomes of their work, and, additionally, the reasons why a respondent may not be using an Agile approach. The determined research questions (RQ) are as follows:

- RQ1: What are the commonly used Agile practices in MSD IT teams?
- RQ2: What Agile practices do MSD IT staff consider most effective?

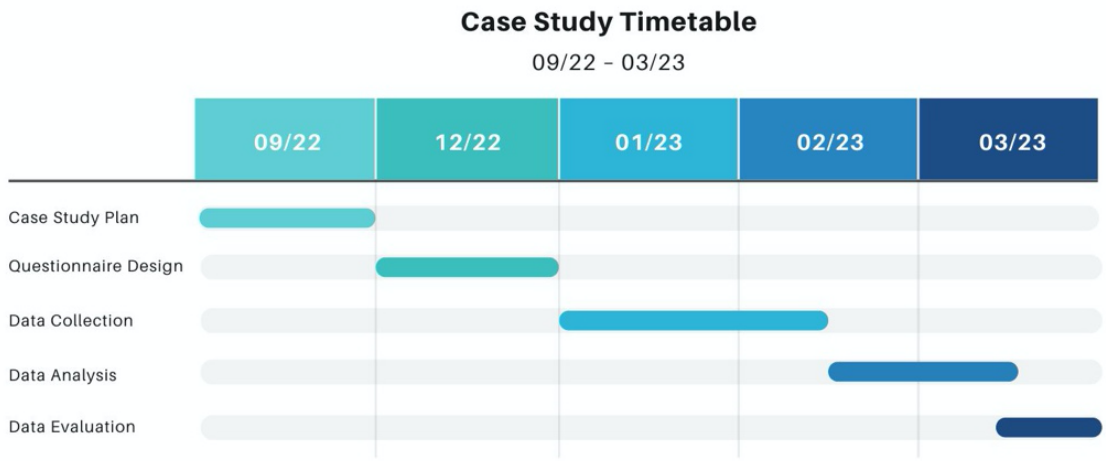
In summary, this study aims to provide a detailed and comprehensive analysis of how MSD IT staff members utilise Agile approaches in their work. The mixed research design, combining both qualitative and quantitative methods, will enable a comprehensive evaluation of the effectiveness of Agile approaches within the company.

As far as the research content is concerned, the case study generally determines respondents' views on Agile-related work processes in their teams. The questions in the questionnaire survey were mostly developed through interviews with members of the Agile coaches' team and team's manager, who all provided feedback on the proposed questions as well as additional suggestions based on their areas of interest and the value that MSD could gain from further investigation. Notably, a total of three questions (21.4%) were authored directly by the team manager. In addition, insight from other sources and studies also served as an inspiration for additional questions.

The preparation phase involved the comprehensive arrangement of the thesis writing and case study process in collaboration with MSD, including the company's Senior Specialist Business Consultant, MSD's attorney, and my manager. The case study plan was developed with the Agile development team and team manager through both in-person and virtual meetings. The preparation phase encompassed a variety of activities, such as clarifying MSD's policy standards for writing the thesis, obtaining approval and consent from my manager for the case study proposal, and engaging in discussions with the Agile development team and its manager to determine the necessary data to be obtained, leading to the development of the questionnaire's questions. As part of the preparation phase, a pilot study was implemented as part of the pre-survey to verify the accuracy, clarity, and appropriateness of the wording of the questions in the planned questionnaire on a small sample of respondents. As a result, the developed questionnaire was first distributed to my manager and teammates from the Agile coaches' team, who attempted to complete the questionnaire themselves. During this process, they scrutinised the set questions and answer options in greater detail, as well as the duration required to complete the questionnaire, and subsequently provided feedback, suggestions, and further advice and comments.

A research timetable was developed as a basis for preparation, illustrated in *Table 4* below, which shows the duration of each stage of the case study.

Table 4 Case Study Timetable



Source: own processing

The timetable illustrated above shows the sequence of events for the MSD case study, which commenced in September 2022, coinciding with the commencement of the present thesis and following approval from both MSD and my supervisor. The process of developing and refining the survey questionnaire is slated to occur in December 2022, with the aforementioned pilot study scheduled to conclude by the end of the month, which will allow for necessary modifications to be made to the questionnaire. It was fairly anticipated that acquiring an adequate number of responses would be challenging; thus, the data collection phase is expected to take approximately one and a half months, from January 2023 until mid-February 2023. The chosen months were selected to maximise the number of respondents, as the majority of MSD employees typically have more time available and are better able to concentrate after the New Year and holiday season. Following the data collection, data analysis and evaluation of the findings will occur in late February and March 2023, respectively.

Concerning specifically the quantitative research methodology in the form of a questionnaire survey, the statistical population under investigation comprises employees within the MSD IT division who are involved in the biopharmaceutical company. The sample selection of participants utilised a non-probability sampling technique, convenience sampling, due to the feasibility of reaching respondents at a given time and virtual space, as well as their willingness to participate in the quantitative research. The choice to survey employees from the IT division was based on the relevance of the case study to their work, given the expected use of Agile techniques in MSD IT teams.

During the implementation phase, the primary data collected through the quantitative questionnaire research was based on an electronic form of the questionnaire survey of MSD IT employees, and various communication channels were utilised to distribute the questionnaire.

Firstly, the questionnaire was sent to selected respondents via internal communication tools such as Microsoft Outlook and Microsoft Teams. Next, it was also distributed by my manager and colleagues at Agile coaches' meetings as well as shared on MSD platforms and private cloud-based social networks like Yammer. In relation to the issues encountered during the implementation phase, it was initially anticipated that the process of data collection would proceed at a more expedited pace than what was actually observed. Consequently, the responses were not gathered as promptly nor in the magnitude that had been previously anticipated. In addition, the main contact person who has given me access to all required resources has been my manager; nonetheless, as mentioned previously, the team of Agile coaches have also provided assistance with communication, particularly with respect to the distribution of the questionnaire.

In conclusion, the practical part of the thesis is devoted to the following analysis of the data obtained from the questionnaire survey, which will also be briefly compared with the latest annual State of Agile Report (2022), published by Digital.ai. Subsequently, the empirical study is devoted to evaluating and commenting on the results and proposing recommendations for MSD based on the data obtained from the questionnaire survey and interviews with accredited Agile coaches.

6.2 Data Collection Analysis

The present case study employed a questionnaire to gather data from employees of MSD IT, which was created using the web-based tool Microsoft Forms. This tool was selected due to its widespread usage within the organisation for various surveys, as well as its compatibility with MS Excel for data exportation. To ensure that only MSD employees could participate in the study, the questionnaire was restricted to internal access only. In total, 85 responses were collected from the questionnaire survey, of which two responses were removed as these respondents are not part of the statistical population, MSD IT employees; therefore 83 responses will be analysed.

Anonymity was employed in order to ensure the highest level of honesty and accuracy of responses, while adhering to internal policies. The questionnaire was designed to include 14 questions in total, with a combination of different types of questions. Specifically, nine of the questions were semi-closed, two were open-ended, two were scaled using a Likert scale, and one question was closed-ended. It is important to note that not all respondents answered all 14 questions, as certain questions were displayed only to those respondents who practise Agile methodologies, while others were displayed only to those who do not. By tailoring the questionnaire in this manner, it aimed to optimize the quality and relevance of the data gathered. The full questionnaire, containing the questions and possible answers, is shown in the appendices.

The initial segment of the survey was aimed at eliciting identification information from the respondents. This was accomplished through two questions designed to provide a comprehensive profile of the group. The first question was an open-ended question, constructed by the Agile Coach team manager, which probed the general perception of MSD personnel towards the company's adoption of Agile methodologies, as well as its integration into their respective teams. Respondents

were instructed to frame their responses as they would when conversing with their friends. This question was formulated more for MSD top management with the intent of facilitating an individual exploration of the particular views held by MSD staff regarding the Agile practices within the organisation.

Based on the responses to the question, which explored the respondent's general view of the Agile approach in MSD and how it is going in their teams, it seems that the most common pattern among respondents is that MSD is still in the process of adopting and implementing Agile methodologies. While some teams have made progress and are using Agile effectively, there are still many teams and divisions that are in the early stages of implementation, and there is a varying degree of understanding and usage across the company. Many respondents indicate that there is still a learning curve, and that the organisation is on a journey towards becoming more Agile. Additionally, some respondents note that there is resistance or confusion from some team members or leaders, but there is also a recognition that Agile has the potential to improve project outcomes, teamwork, and communication. Overall, it seems that MSD is striving to be more Agile, but there is still work to be done to fully integrate Agile methodologies across the organisation.

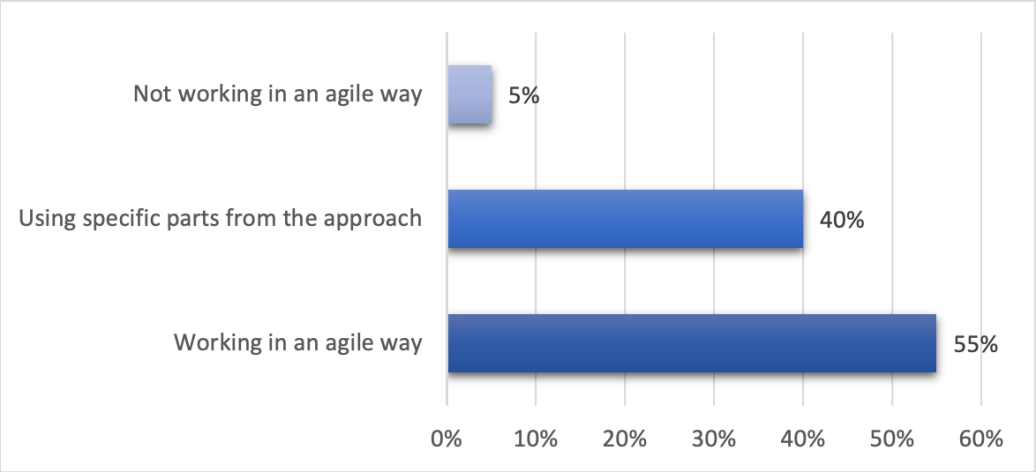
Next, trichotomous question aimed to investigate whether the respondents' respective teams adopted an Agile methodology. Respondents were presented with a closed-ended question and given three options to choose from: (a) yes, they are working in an Agile manner, (b) they use specific parts from the Agile approach, or (c) no, they do not implement Agile practices at all. The response to this inquiry was determined by subsequent questions that were contingent on the respondent's initial response. In the case that a respondent indicated that they utilise Agile methodologies, or at least partially, they were prompted to respond to the survey question using Agile methods/frameworks. Conversely, if a respondent reported that they do not employ Agile methodologies in their team, they were directed to the question that explored the reasons for not using the Agile approach, which was only relevant to respondents who did not utilise Agile methodologies since the purpose of 10th Question was to ascertain the reason for not using the Agile approach.

Based on the graphical representation by column chart on the next page *(Graph 1)*, the findings pertaining to the responses indicate that a majority of the respondents, amounting to 46 MSD IT employees or 55% of the sample, work in an Agile way. Conversely, 33 MSD IT staff members or 40% of the respondents reported using specific parts from the Agile approach. A minority of only 4 respondents (5%) of the total sample do not work in an Agile way at all.

These findings are consistent with the continued rise in the popularity of Agile methodologies in the IT area. According to the 15th Annual State of Agile Report by Digital.ai (2021), 98% of survey respondents reported using Agile approaches, while only 2% reported using waterfall methodologies. In addition, a recent survey by the DevOps Institute (2021) found that 86% of organisations reported using Agile methodologies in some capacity, and 58% reported that they were increasing their investment in Agile approaches in the coming year. These data points indicate that Agile methodologies will likely continue to be a dominant force in the IT industry for the

foreseeable future. In light of these trends, it is not surprising to find that a majority of MSD IT staff members work in an Agile way, as reflected in the results illustrated in the graph below *(Graph 1)*.

Graph 1 Share of MSD IT employees (not) working in an Agile way, n = 83



Source: own processing

Further question was a multiple-choice enumeration question that aimed to explore the Agile frameworks and methods employed by the IT staff of MSD who at least partially use Agile in their teams. According to the 15th Annual State of Agile Report, Scrum and Kanban remain the most widely used Agile methodologies globally. The State of Agile Report found that Scrum was used by 72% of respondents, while Kanban was used by 42%. This study confirmed the claim that the most popular Agile frameworks are Scrum and Kanban, with Scrum being the most widely used. In the context of MSD's IT staff, as depicted in the bar chart *(Graph 2)*, the results indicate that Scrum was reported by 83.5% of respondents, while Kanban was used by 62% of the participants, which aligns with the global trends. The Lean approach was ranked third; however, it exhibited a significant decline of over 40% in comparison to the first two methods, garnering a total of 17.7% of the responses.

The study reported that 42 respondents (53.2%) indicated the use of both Scrum and Kanban methodologies. This finding suggests the possibility of various situations, such as respondents working with multiple teams employing different frameworks or working in a single team that integrates different techniques from Scrum and Kanban. It is common practice to use Kanban boards to monitor the progress of a sprint in a Scrum team. Moreover, the question arises as to why some respondents selected more than one framework, which could indicate working with multiple teams or recognising overlapping similarities between the frameworks. Additionally, the Less is essentially several teams working in Scrum, with communication between them. As a result, one can argue that using Less also entails using Scrum; thus, theoretically, all three respondents who selected Less also reported using the Scrum framework.

Of particular interest is the observation that some MSD IT teams deploy their own customised solutions, with nine participants (11.4%) acknowledging such an approach. Similarly, an

equal number of respondents (9 employees) selected the "other" option, where they specified the names of their own custom Agile methodologies. Several of those respondents listed various frameworks such as "disciplined Agile", "combination of good practices from multiple frameworks", and "OKR's", which stands for Objectives and Key Results, among others.

The study also revealed a surprising and unexpected finding that although a small proportion of respondents, comprising 6.3%, some respondents reported using the Scaled Agile Framework (SAFe) methodology, despite its official discouragement. This observation is remarkable as it raises questions as to why individuals would openly admit to using a methodology that is not recommended and suggests the need for further research to explore the factors underlying such a practice.

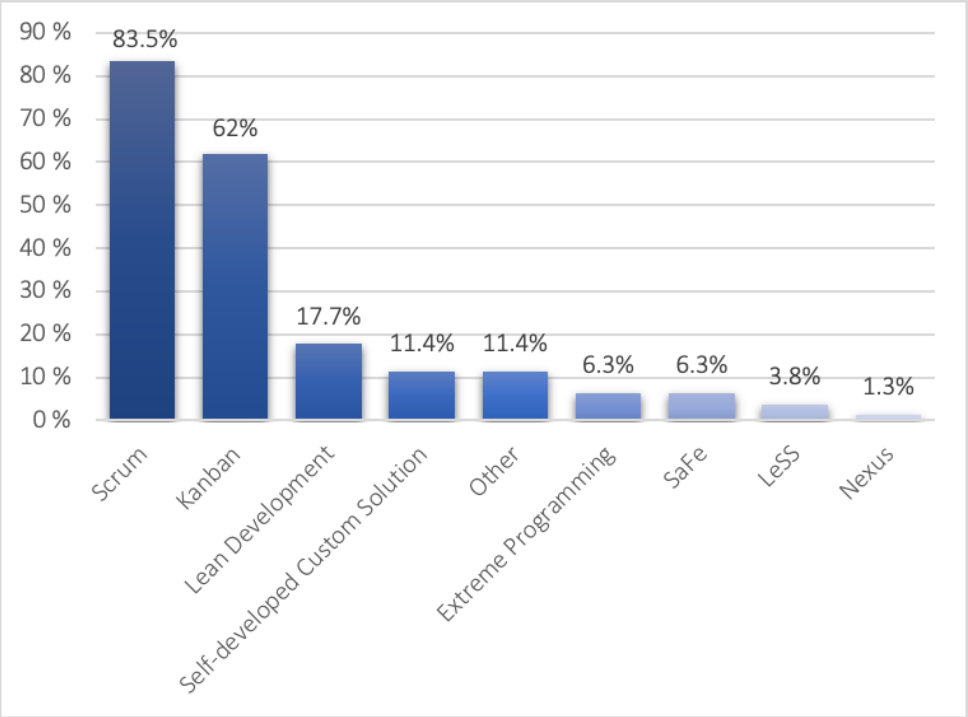
Finally, the study found that Extreme Programming (XP) was not a popular Agile framework, with only seven respondents (6.3%) reporting its use. This finding suggests the possibility of multiple reasons for its limited use, as XP is often considered a development practice rather than an Agile framework. Further investigation is required to determine whether XP is indeed not employed, which could be a concern given its many beneficial development practices, or if the respondents who did not report its use are from less technical roles and lack visibility into how XP is leveraged in technical teams. Additionally, this observation is supported by the finding in the following question, which explored what Agile practices MSD IT staff consider as most effective, where none of the respondents selected technical excellence as one of the three most useful Agile approaches.

In conclusion, this study aimed to explore the Agile frameworks and methods used by IT staff at MSD who partially employ Agile in their teams. The results of the study confirmed that Scrum and Kanban are the most widely used Agile methodologies, consistent with global trends. The study also revealed that most respondents reported using more than one methodology, indicating the possibility of MSD IT teams combining different methods/frameworks. The Lean approach ranked third, exhibiting a significant decline in comparison to Scrum and Kanban. Notably, some MSD IT teams deploy their own customised solutions.

These findings align with the trend of organisations tailoring Agile practices to fit their specific needs, as highlighted in the 2020 Agile Trends Report by Agile Alliance. The report also notes the rise of hybrid approaches, where organisations combine different Agile frameworks and methods to create a unique solution that suits their context. Overall, the findings of this study suggest that MSD's IT staff mostly utilise well-established Agile methodologies, primarily Scrum and Kanban. Further research would be useful to explore the factors underlying such practices, including the limited use of Extreme Programming (XP) and the open use of SAFe, which are not widely recognised Agile frameworks.

The present study displays the outcomes of the data analysis pertaining to the question researching the most commonly used Agile methods/frameworks through a bar chart *(Graph 2)* that is presented on the subsequent page.

Graph 2 Share of MSD IT employees using specific Agile methods/frameworks, n = 79



Source: own processing

The following analysed questions are fundamental research questions, consequently, will be subject to thorough analysis. In the research questions, MSD IT staff members were asked about the Agile practices their teams utilise and which ones they consider to be the most productive. Given their interrelated nature, these questions will be juxtaposed within a contingency table to facilitate a more comprehensive comparison.

The former research question sought to explore what Agile practices MSD IT teams use. The question was presented in the form of a multiple-choice item in the questionnaire, aimed at ascertaining the employment of specific Agile practices by the MSD IT personnel within their teams. To ensure ease of completion, the question featured an ample selection of over 15 possible responses. An additional option for "others" was also included to account for any Agile practices that were not explicitly stated. Respondents were allowed to select any number of Agile practices that they use in their teams.

Results of an analysis of the responses to the question that aimed at investigating the use of Agile practices among MSD IT employees, indicated that a majority of employees utilise the Agile practice of product backlog. The prioritised list of user requirements that the development team uses to guide the development of the product is used by 66 MSD IT employees (83.5%). It was to be expected that product backlog would be one of the main used Agile practices since it is one of the key elements of a popular framework, Scrum methodology. Anyway, it is a positive outcome as the well-maintained product backlog ensures that the team is focused on delivering value to the customer and that it helps in planning and prioritising work items.

In conjunction with having a product backlog, 64 respondents (81%) reported having a Product Owner in their team, indicating its widespread use, who is responsible for maximising the value of the product and the work of the development team. The Product Owner achieves this by creating and maintaining a product backlog, ensuring that the team understands the items in the product backlog, and making decisions about the release of the product. This is mentioned primarily because, interestingly, the results showed that 84% of employees have a product backlog, but 81% have a Product Owner, indicating that in some teams, the Product Owner is missing. This can lead to confusion and misalignment within the team and can affect the quality of the product being developed.

The third most used Agile practice was the use of Scrum or Kanban board, utilised by a total of 61 (77.2%) MSD IT employees. It is right-thinking since visual management tools like Scrum and Kanban boards help teams in managing their work by providing a clear visual representation of the work items and their progress. These tools also facilitate communication and collaboration among MSD IT team members and help in identifying and resolving bottlenecks. However, it is unusual that not every respondent who uses Kanban method working in Scrum/Kanban board, a basic tool of the method; 6 respondents out of 50 using Kanban do not use a Scrum/Kanban board. Additionally, a similar number of employees reported having a Scrum Master in their team, with a total of 58 respondents (73.4%) reporting their own. The Scrum Master is responsible for facilitating Scrum events, removing impediments that are blocking the team's progress, and ensuring that the team is following Scrum principles and practices. Having a skilled Scrum Master is crucial for the success of Scrum implementation in teams; hence, it was also surprising to note that more respondents reported having a Product Owner than a Scrum Master. This could be because some teams may not have a clear understanding of the roles and responsibilities of a Scrum Master and may not see its importance in Agile practices.

Other Agile practices utilised by the majority of respondents on their team, over 50% of employees, include retrospectives (69.6% of respondents), sprint planning (68.4%), working in iterations (64.6%), daily stand-ups (57%), small self-organised teams (55.7%), feedback loops (54.4%) and working together daily (53.2%). Nevertheless, these practices should be implemented more still as these Agile practices are highly recommended by the Agile Manifesto and are widely used in Agile development. For example, working in iterations is a basis of Scrum approach and the unusual is that it is not used by every respondent who reported using of Scrum framework; only 46 respondents using Scrum work in iterations. Besides, insufficient conformity to the practice of daily collaboration may be attributed to the proclivity of the organisation to form global teams with incomplete allocations, thereby preventing physical co-location.

Only a few MSD IT employees (3 in total) chose the answer "other", where respondents reported having a team manifesto, using collaboration both virtually and in person, and one of them noted separating between capability teams and cross-functional teams type of work. To clarify, a team manifesto is a document that outlines the team's values, goals, and working principles that helps in aligning team members and setting expectations for how the team will work together. Collaboration, both virtually and in person, is a crucial aspect of Agile practices, and teams

should strive to use tools and technologies that support collaboration and communication. Separating between capability teams and cross-functional teams type of work is a strategy that some organisations use to structure their teams based on their skill sets and expertise. Capability teams focus on specific capabilities or technologies, while cross-functional teams have members with diverse skills and expertise to deliver a product or service.

Next, the practice of conveying information face-to-face was reported by relatively few respondents (23 respondents), which is a vital aspect of Agile practices and should occur more frequently. According to the Agile Manifesto, face-to-face communication is the most effective way of conveying information within a development team. This is because it facilitates quick feedback, collaboration, and problem-solving. Furthermore, MSD IT staff should practice more sprint planning, daily stand-ups, and review to examine team's effectiveness. The team's planning of the work to be done in the upcoming sprint, daily meetings where team members discuss progress, blockers, and plans for the day and meetings held at the end of each sprint to review the work done and gather feedback from stakeholders are important Agile practices which help in improving team communication, collaboration, and productivity.

At the same time, if Agile methods are used, and the team wants to be genuinely Agile, it should always consist of small self-organised teams and involve the customer as much as possible in the process to find out as much information about the customers as possible. This is because self-organised teams are more efficient and effective in delivering products and services, and customer needs discovery in the process ensures that the team is building a product that meets the customer's needs and expectations. Therefore, these Agile practices should occur more frequently and be considered important.

The results of the research are generally consistent with other studies, reports, and articles related to Agile practices. For example, the 15th and 16th State of Agile reports also highlight the widespread use of product backlog, Scrum and Kanban boards, retrospectives, and sprint planning in Agile development. However, the research also highlights some areas where Agile practices may not be fully implemented or understood by MSD IT employees, such as the relatively low reported use of face-to-face communication and the discrepancy between the number of employees with a Product Owner versus a Scrum Master.

In conclusion, the analysis of responses to the question identifying using Agile practices has provided valuable insights into the use of Agile practices among MSD IT employees. Most employees use the Agile practice of product backlog, followed by having a Product Owner in their team and the use of Scrum or Kanban board. While these are positive outcomes, there were also some areas for improvement, including the need for more face-to-face communication, increased use of sprint planning, daily stand-ups, and review, and the importance of having a skilled Scrum Master. The results also indicate that not many respondents engaged in the Agile practice "customer need discovery", and prioritised outcome orientation. These findings shed light on the prevalent disparity between the IT industry and business practices. The results highlight the

importance of effectively implementing more Agile practices to ensure better team communication, collaboration, and productivity.

The outcomes encompassing the number and share of Agile respondents employing distinct Agile practices are presented in the tabular format below *(Table 5)*.

Table 5 MSD IT employees using specific Agile practices in their teams, n = 79

Agile Practice	Number of respondents	Share of respondents
Product Backlog	66	83.5%
Having Product Owner	64	81%
Kanban board/Scrum board	61	77.2%
Having Scrum Master	58	73.4%
Retrospectives	55	69.6%
Sprint Planning	54	68.4%
Working in iterations	51	64.6%
Daily Stand-ups	45	57%
Small self-organised teams	44	55.7%
Feedback loops	43	54.4%
Working together daily	42	53.2%
Examine team's effectiveness (Review)	36	45.6%
Frequent product delivery	35	44.3%
Customer need discovery	35	44.3%
Outcome oriented prioritisation	31	39.2%
Convey information face to face	23	29.1%
Technical agility (XP practices)	15	19%
Other	3	3.8%

Source: own processing

Lastly, what is interesting to note is that among the 66 respondents who professed utilisation of Scrum, a mere 53 reported presence of a Scrum Master in the team, while 56 affirmed the existence of a Product Owner. Notably, only 19 out of the 66 respondents confirmed the employment of the complete set of Scrum-related meetings (daily meetings, planning, review, retro), hence indicating suboptimal utilisation of the framework.

The latter research question has been related to the previous research question, which also inquired through a multiple-choice format. Nevertheless, the number of possible selections was restricted, with an MSD IT employee allowed to select a maximum of three responses, which a respondent comprised as the three most effective Agile practices. The options available for selection were identical to those presented in the preceding question for ease of reference. In the event that none of the given options was suitable, respondents were provided with the opportunity to indicate an "other" category and specify their chosen Agile practice accordingly.

The following analysis will be from the question that investigated the effectiveness of various Agile practices among MSD IT staff. The data results from this question, which asked participants to select the most effective Agile practices, revealed that having a Scrum Master was considered

the most effective practice, selected by a total of 22 respondents. Further, Agile practices as the presence of a Product Owner, daily collaboration, and product backlog, were deemed equally effective, with 20 respondents choosing each practice. Therefore, in the data collected has been emphasised the importance of having a Scrum Master for the successful implementation of Agile practices. The finding that having a Scrum Master is considered as the most effective practice is a positive outcome, as having a proper Scrum Master is crucial to implementing Agile effectively. According to a report by Scrum.org, a certified Scrum Master plays a critical role in ensuring the Scrum team follows the Scrum framework properly and removing any impediments that could hinder the team's progress. Furthermore, the equal effectiveness of Product Ownership and product backlog suggests that these practices are also essential for Agile success. This finding that having a Product Owner and product backlog is equally effective also aligns with the Scrum framework, which emphasises the importance of these roles in the development process.

The results of this question also found that Agile practices "working in iterations" and "customer need discovery" were deemed effective by 16 respondents each. These responses are consistent with the Agile Manifesto, which emphasises the importance of customer collaboration and responding to change over following a plan. Similarly, retrospectives and feedback loops were deemed equally effective according to 13 respondents, which aligns with the Agile principle of continuous improvement.

In contrast, technical Agility (XP practices) was seen as the least effective; despite 15 respondents (19%) reporting its use in their teams, no one selected this Agile practice as one of the most effective. The Agile practices of frequent product delivery (selected by 5 respondents) and conveying information face to face (selected by 8 respondents) were also chosen by a few respondents, which respectively suggests that these practices may need more attention. Agile methodology advocates for delivering working software frequently and encouraging face-to-face communication as it promotes better collaboration and a shared understanding of the project's goals. In addition, the next Agile practice “outcome-oriented prioritisation” was considered one of the most effective practices by 11 respondents.

Interestingly, the data found that the Scrum board or Kanban board, which are fundamental Agile practices used by many MSD IT teams, were not selected by as many respondents as expected, with only 12 employees deeming them most effective. Generally, these boards are widely used in Agile software development; however, this could indicate that other visual management techniques are being used instead. Additionally, the two remaining Agile practices, “small self-organised teams” and “examine team's effectiveness (review)” were chosen by only 9 respondents each.

It is important to note that not all respondents chose three Agile practices, but some selected only one or two. Some participants even added other practices and comments to the "other" option for this question, such as self-accountability, co-ownership, and automated testing and CICD pipelines. Automated testing is the use of software tools to execute tests and compare actual results with expected results. This type of testing is used to automate repetitive and time-

consuming tasks, such as regression testing, and it helps to ensure that software applications are functioning as expected. Continuous Integration/Continuous Deployment (CI/CD) pipelines are a set of processes and tools that automate the building, testing, and deployment of software applications. The goal of a CI/CD pipeline is to enable teams to deliver software quickly and reliably by automating the entire process from code commit to production deployment. Moreover, automated testing plays a critical role in the CI/CD pipeline by providing feedback on the quality of the code changes before they are deployed to production. Respectively, it can be said that these responses suggest that there may be additional practices that are effective for certain MSD IT staff and their teams or situations.

In summary, the results of the analysed question about the most effective Agile practices provide valuable insights into the effectiveness of various Agile practices among MSD IT employees. The findings suggest that having a Scrum Master, Product Ownership, and product backlog, and working together daily are essential practices for implementing Agile effectively. Additionally, the study highlights the importance of considering individual team needs and circumstances when selecting Agile practices. This aligns with the Agile Manifesto's principle of valuing individuals and interactions over processes and tools. The study also underscores the importance of considering individual team needs and circumstances when selecting Agile practices. This aligns with the Agile Manifesto's emphasis on responding to change and prioritising customer collaboration, as it recognises that different teams may have different needs and priorities that require different approaches to Agile development.

The tabulated data *(Table 6)* on the subsequent page delineates the precise number and share of respondents working in an Agile way, with specific Agile practices being considered by the number/share of respondents among the three most effective.

Table 6 MSD IT employees considering specific Agile practices as most effective, n = 79

Agile Practice	Number of respondents	Share of respondents
Having Scrum Master	22	27.8%
Having Product Owner	20	25.3%
Working together daily	20	25.3%
Product Backlog	20	25.3%
Working in iterations	16	20.3%
Customer need discovery	16	20.3%
Sprint Planning	14	17.7%
Daily Stand-ups	14	17.7%
Retrospectives	13	16.5%
Feedback loops	13	16.5%
Kanban board/Scrum board	12	15.2%
Outcome oriented prioritisation	11	13.9%
Small self-organised teams	9	11.4%
Examine team's effectiveness (Review)	9	11.4%
Convey information face to face	8	10.1%
Frequent product delivery	5	6.3%
Other	4	5.1%
Technical agility (XP practices)	0	0%

Source: own processing

Regarding the comparison of the data obtained from the question that explored using Agile practices and the question identifying the most effective ones, as previously mentioned, most people use the Agile practice of product backlog, while having a Scrum Master is considered the most effective Agile practice. However, what is still striking is that out of the 58 MSD IT employees who have a Scrum Master, only 22 of them selected it as one of the most effective Agile practices, which is not even half. This suggests that there may be a gap between the perceived and actual effectiveness of the Scrum Master role. Yet the Scrum Master in the team is seen as the main role of Agile methods. Similarly, although product backlog is used by 66 MSD IT employees, just under one-third (20 respondents) selected it as one of the most useful practices. This could be due to a lack of understanding about the benefits of product backlog or a lack of effective implementation.

Further, it is surprising that although many respondents use Scrum/Kanban boards (61 in total), only 12 respondents consider it one of the most effective Agile practices. This may indicate that Scrum/Kanban boards are being used primarily as a visual aid rather than as a tool to improve team collaboration and productivity. Additionally, although 64 MSD IT employees have a Product Owner, only 20 times this Agile practice was chosen as one of the most effective, which is also quite a big difference. This could be due to a lack of clarity about the role of the Product Owner or a lack of effective communication with the rest of the team.

There is also a big difference in the fact that 55 respondents use retrospectives, but less than one-quarter (13 respondents) consider retrospectives to be significantly effective. This could be due to a lack of understanding about the purpose of retrospectives and its importance; however, it is important to note that retrospectives are a key practice in the Agile methodology, which

emphasises flexibility, collaboration, and continuous improvement. As was previously mentioned in the theoretical part of the thesis, the purpose of a retrospective is to facilitate team learning and continuous improvement by encouraging open and honest communication and identifying opportunities for improvement. Therefore, this Agile practice might be connected with other Agile practices like feedback, review or conveying information face to face. Furthermore, it is also shocking that not a single respondent considers technical Agility (XP practices) as one of the most practical Agile processes, even though 15 MSD IT employees use technical Agility. This could be due to a lack of understanding about the technical Agility or findings these practices are difficult to adopt, or the XP practices can be time-consuming and require a significant investment in training and tools. Some people may question whether the benefits of adopting these practices justify the cost. Moreover, the XP practices are most effective when applied to software development projects that have a high degree of uncertainty or complexity, so some people may argue that these practices are less useful for projects that are more straightforward or predictable.

On the other hand, what is positively surprising is that even though not as many MSD IT employees work together daily (53.2%), compared to the other Agile practices, this technique is considered one of the three most effective. This suggests that "working together daily" is an underutilised but effective practice. As for the Agile practice of conveying information face to face, it is sadly surprising that a small proportion of employees use this Agile practice (29.1%), and an even smaller proportion consider it one of the most effective (10.1%). This may be due to a lack of opportunities for face-to-face communication or a preference for other communication methods. Nevertheless, conveying information face to face emphasises the importance of direct communication among team members in the Agile development process. The idea behind this practice is that face-to-face communication helps to reduce misunderstandings, encourages collaboration and creativity, and leads to better outcomes. It is especially important for distributed teams, such as MSD teams, where communication can be challenging due to physical distance and time zone differences. By using video conferencing tools, team members can feel more connected to each other and communicate in a more natural and efficient way.

However, it is important to note again that there was no limit to the number of responses for the question, which asked what all practices MSD IT employees use, whereas for the other question, which explored the most effective Agile practices, respondents could select a maximum of three Agile practices they considered the most effective. Thus, just because a particular Agile practice was not selected as one of the most effective does not mean that it is considered ineffective.

Overall, the findings of these two questions suggest that there is room for improvement in the implementation and communication of Agile practices among MSD IT employees. According to a study conducted by McKinsey & Company, Agile practices can improve project success rates by 60%, reduce time to market by 30%, and increase team productivity by up to 40%. This highlights the potential benefits of Agile practices in the MSD workplace. Comparing other studies to the present study conducted among MSD IT employees, it is clear that while some of the most commonly used Agile practices are consistent across studies, there is variation in their perceived

effectiveness. This suggests that effective implementation and communication of Agile practices may vary across organisations and industries, highlighting the need for continuous improvement and adaptation. Additionally, there are several cases where individuals have identified practices, they consider most useful but do not implement themselves, warranting further investigation into the reasons underlying this phenomenon.

In order to facilitate a more lucid comparison between the data obtained, the contingency table *(Table 7)* is shown beneath, which illustrates the number of respondents using the specific Agile practice (X) and the number of respondents considering the specific Agile practice to be one of the most effective (Y).

Table 7 Comparison of number of respondents using the Agile practice vs. considering the practice as most effective, n = 79

Name of the agile practice	X	Y
Convey information face to face	23	8
Customer need discovery	35	16
Daily Stand-ups	45	14
Examine team's effectiveness (Review)	36	9
Feedback loops	43	13
Frequent product delivery	35	5
Having Product Owner	64	20
Having Scrum Master	58	22
Kanban board/Scrum board	61	12
Other	3	4
Outcome oriented prioritisation	31	11
Product backlog	66	20
Retrospectives	55	13
Small self-organised teams	44	9
Sprint Planning	54	14
Technical agility (XP practices)	15	0
Working in iterations	51	16
Working together daily	42	20
Summary	761	226
X = Number of respondents using the agile practice		
Y = Number of respondents considering the practice as most effective		

Source: own processing

In the following question, the focus was on exploring the Agility supporting tools that are utilised by MSD IT teams. The question was designed as a multiple-choice question with respondents being allowed to choose an unlimited number of options. The answer choices encompassed a range of commonly recognised Agile tools such as MURAL and JIRA, as well as widely used Microsoft communication tools such as MS Excel, MS Power Point or MS White Board. Notably, the response options also included some Agile tools that are not recommended for use in MSD IT teams, due to their non-internal nature and potential risks associated with exposure of sensitive and confidential corporate data, to determine whether these tools are even being used. As with most questions, an "other" option was provided as well, enabling respondents to report any other Agile tools that they use within their respective teams.

The analysis of the question asking about the Agility supporting tools used revealed interesting insights regarding the utilisation of Agile supporting tools by MSD IT teams. It was observed that the most widely used Agile tool is effective messaging tool MS Teams, which is employed by a significant proportion of the respondents (86.1%). The second and third most commonly utilised Agile tools by MSD IT staff are JIRA selected by 81% of respondents, and MURAL, which selected 79.7% of respondents. These results are consistent with other studies that have highlighted the popularity of JIRA and MURAL in Agile software development projects. According to the last Agile State of the Industry Report, JIRA remains the most widely used Agile tool in the industry, with 50% of respondents using it, and MURAL is also a popular tool still. The state of Agile report also highlights that Microsoft Teams has become a more popular collaboration tool among Agile teams, with a 30% increase in its usage from the previous years, which is also related to the results of the MSD Case Study.

As was mentioned, this finding is consistent with other public studies which have reported the increasing popularity of communication and collaboration tools in Agile software development projects. The State of Agile report also highlights that many organisations are using multiple Agile tools, with 44% of respondents using two or more Agile tools. This finding is consistent with the results of the current study, where several Agile tools were found to be used by MSD IT teams. Most MSD IT staff chose three or more Agility supporting tools in use. Only a few respondents (3 respondents in total) chose only one tool, the Agile tool JIRA, and a total of 11 respondents (13.9%) selected two tools, mostly containing JIRA/JIRA Align, MS Teams, or MURAL.

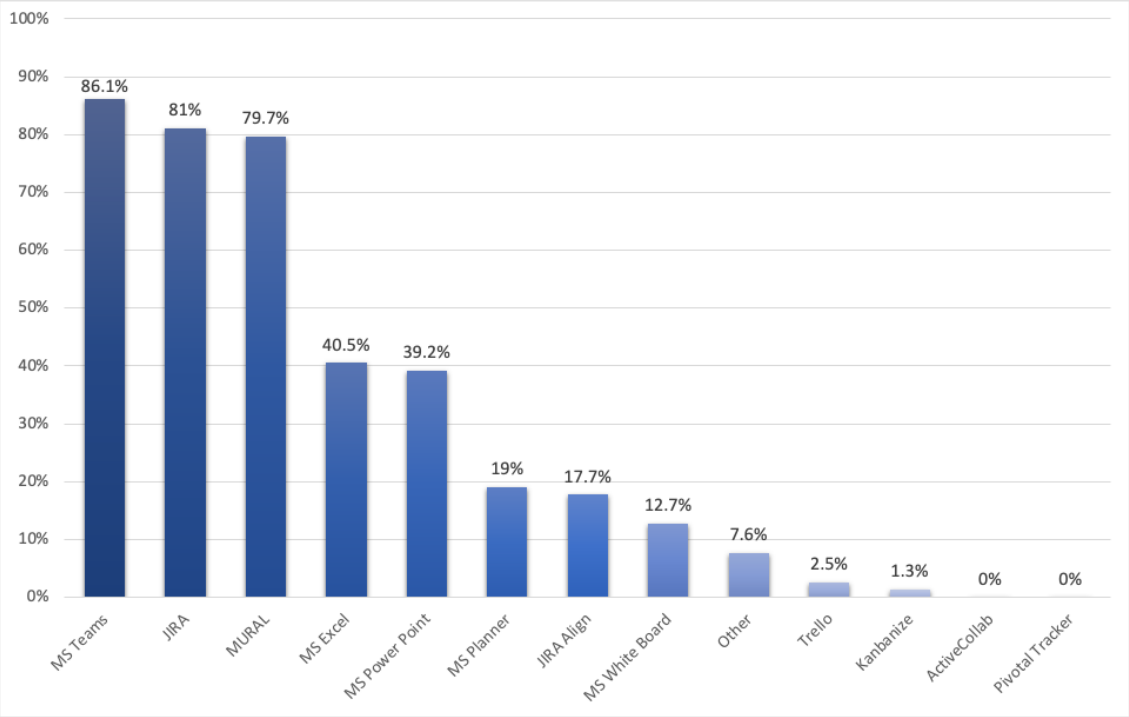
Interestingly, Kanbanize and Trello were also selected by some respondents despite being unsupported Agile tools in MSD IT teams, even though a small number of respondents (3.8%). However, other unsupported Agile tools, such as ActiveCollab and Pivotal Tracker, were not utilised by any of the respondents. In addition, a few respondents (7.6%) opted for the "other" option, with two of them mentioning Azure Boards as their used Agility supporting tools, which is a stand-alone tool included in the Azure DevOps suite that enables teams to plan, track, and discuss work across the entire software development process. Other tools mentioned by the respondents include Confluence, which is Atlassian's cloud-based web application for online collaboration; next JIRA Roadmap, which enables breaking down large jobs into epics and assigning specific dates and deadlines, and Quantive, an OKR tracking software developer. One of the respondents who chose the "other" option reported that their team uses custom-made survey tools and real-life whiteboards, including sticky notes.

In conclusion, the results of the analysis of the question to determine the Agile tools used shed light on the Agile supporting tools that are being used by MSD IT teams. The findings reveal that the most commonly utilised tool is MS Teams, followed by JIRA and MURAL, which are consistent with industry reports. It is noteworthy that respondents also selected some unsupported Agile tools, indicating that the use of such tools cannot be probably completely eliminated. However, most MSD IT staff use three or more Agility supporting tools, indicating the significance of the Agile methodology in their work. The inclusion of an "other" option also enabled the identification of some unique tools being utilised by the teams, such as custom-made survey tools

and real-life whiteboards. These findings provide valuable insights for the organisation to enhance its Agility by supporting tools and underscore the need to maintain a balance between utilising industry-standard tools and incorporating unique tools based on the MSD IT team's needs.

The outcomes of the data analysis from the question pertaining to the utilisation of Agility supporting tools, are depicted graphically below *(Graph 3)*.

Graph 3 Share of MSD IT employees using Agility supporting tools, n = 79



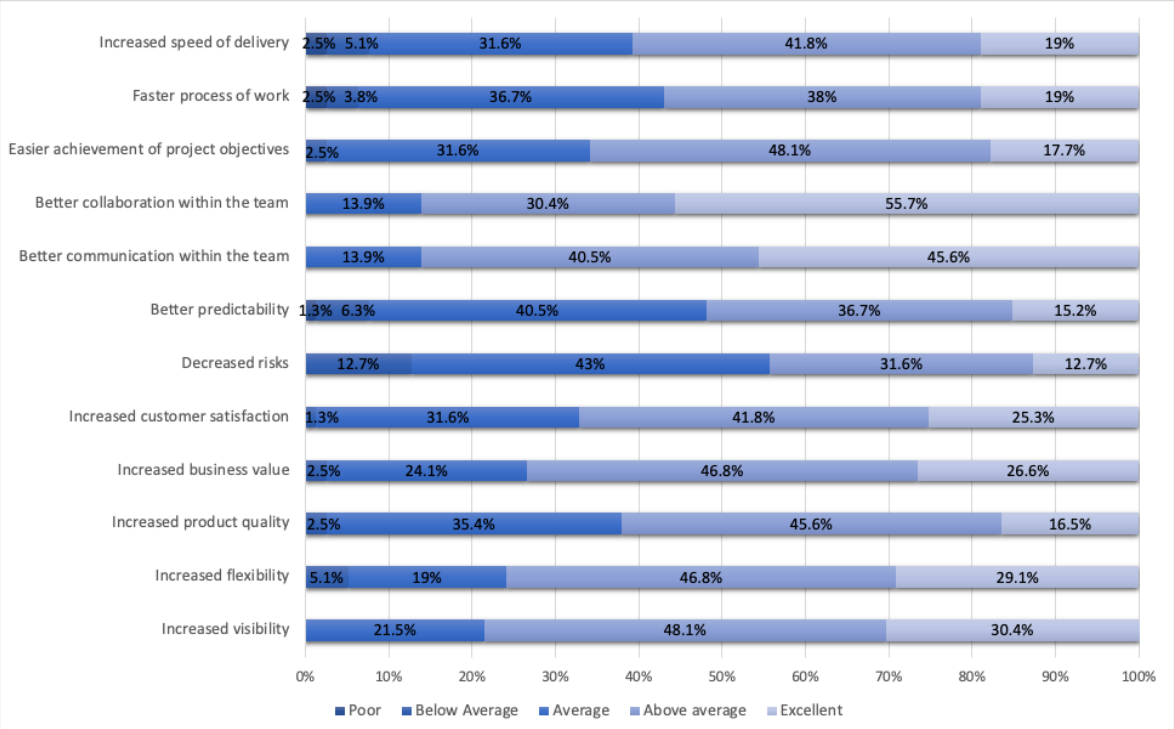
Source: own processing

Overall, the results illustrated in the graph above (*Graph 3)* suggest that MSD IT teams employ a range of Agile supporting tools for communication, collaboration, and project management, with MS Teams, JIRA, and MURAL being the most commonly utilised tools. The selection of unsupported Agile tools by some respondents highlights the need for clear guidance and training on the recommended tools for use in MSD IT teams. Future research could explore the effectiveness of different Agile tools and how they contribute to the success of Agile various types development projects. What is important is selecting the right tools for an organisation's specific needs rather than just using the most popular ones. The suggestion is that MSD IT teams should consider factors such as team size, complexity of projects, and desired level of collaboration when selecting Agile tools. MSD IT teams were found to be using a range of Agile tools, including some non-supported ones, depending on their specific needs and preferences. Anyway, the overall analysis of the results on the use of Agile tools shows that MSD IT teams use Agile tools during their work and also combine them. In addition, the adoption of Agile practices and remote collaboration tools, such as video conferencing and virtual whiteboards, may have been influenced by the COVID-19 pandemic.

Next, two of the questions in this study were structured as scaled questions employing the Likert scale methodology to examine and quantify respondents' attitudes. Specifically, the former question aimed to scrutinise the attitudes of MSD IT personnel regarding the diverse advantages of the Agile approach, while the latter question sought to explore the relative significance of factors to the success of implementing an Agile approach and, conversely, to identify factors that were perceived to be unimportant by the participants.

Therefore, this study aimed to examine MSD IT employees' attitudes towards the benefits of adopting an Agile approach. The findings were based on responses to the question that asked about employees' perceptions of the benefits that Agile brings to their teams. In the range of options, there were mentioned several benefits obtained from various studies and other public sources mentioned in the theoretical part of the thesis that an Agile approach should bring. This question was used to determine whether the Agile approach brings these benefits to MSD IT teams and how team members perceive the Agile benefits. For a clearer overview of the results of how the given potential benefits are rated by MSD IT employees, a Likert scale graph *(Graph 4)* is shown beneath.

Graph 4 Share of MSD IT employees ranking specific Agile benefits, n = 79



Source: own processing

Based on the findings of the question aiming to assess the attitudes of MSD IT employees regarding the benefits of using Agile methodology in their teams, which is illustrated in the graph on the previous page *(Graph 4)*, it can be deduced that the majority of employees (55.7%) consider "Better collaboration within the team" to be the most significant – excellent – benefit of Agile methodologies. This result is consistent with the Agile State of DevOps report, 2021, which found that collaboration was the most commonly cited benefit of Agile (DORA, 2021). Other studies have

also identified collaboration as a significant benefit of Agile practices. For example, a study by Pika et al. (2020) found that Agile practices improved collaboration among team members in most of teams. Furthermore, “improved communication within the team” is also seen as an excellent benefit by 45.6% of MSD IT employees. This finding is also consistent with the Agile State of DevOps report, which identifies communication as a key enabler of Agile practices (DORA, 2021). Similarly, a study by Forsberg et al. (2020) found that Agile practices improved communication and reduced misunderstandings among team members in many organisations.

Further, increased visibility is identified as an excellent benefit of Agile by 30.4% of employees; however, most of the respondents (48.1%) consider better visibility as a benefit above average. The rest of the respondents (21.5%) rank better visibility as an average benefit. Other studies have also identified increased visibility as a benefit of Agile practices. Another study, like the Agile State of DevOps report, highlights that visibility is a key enabler of Agile practices (DORA, 2021). Also, a study by Jansen et al. (2020) found that Agile practices improved transparency and visibility into the development process.

Conversely, “increased speed of delivery”, “faster process of work” and “better predictability” were the only ones rated as poor benefits of Agile even though by a low percentage of respondents (a total of 6.3% of respondents). This finding is consistent with the Agile State of DevOps report, which notes that Agile practices may not always result in faster delivery (DORA, 2021). Other studies have also found that Agile practices do not always result in faster delivery or better predictability (Pika et al., 2020; Forsberg et al., 2020).

The benefits of decreased risks and better predictability are considered average benefits by MSD IT employees, with 43% and 40.5% of respondents, respectively, rating them as such. Nevertheless, the Agile State of DevOps report highlights that Agile practices can help reduce risk and improve predictability (DORA, 2021). Other studies have also identified reduced risk and better predictability as benefits of Agile practices (Pika et al., 2020; Forsberg et al., 2020).

In conclusion, the study's findings from the question on the rating of Agile benefits suggest that “Better collaboration within the team”, “increased visibility” and “improved communication within the team” were ranked as the most significant benefits of Agile methodologies for MSD IT employees. These results are consistent with the Agile State of DevOps report and other studies, which identify collaboration, communication, and visibility as key enablers of Agile practices. Namely, the findings of the 16th State of Agile report has been determined that the primary advantages observed by organisations following the adoption of Agile methodologies are enhanced team productivity, increased project visibility, and expedited time-to-market. Conversely, as the least significant benefits were ranked “decreased risks”, “increased speed of delivery” and “better predictability”. It is noteworthy that in the survey conducted among MSD IT employees, none of the potential benefits of Agile methodologies were rated only as below average or only as poor. Instead, each benefit was perceived by some proportion of employees as excellent and above average. This observation indicates that the surveyed employees have a positive perception of the

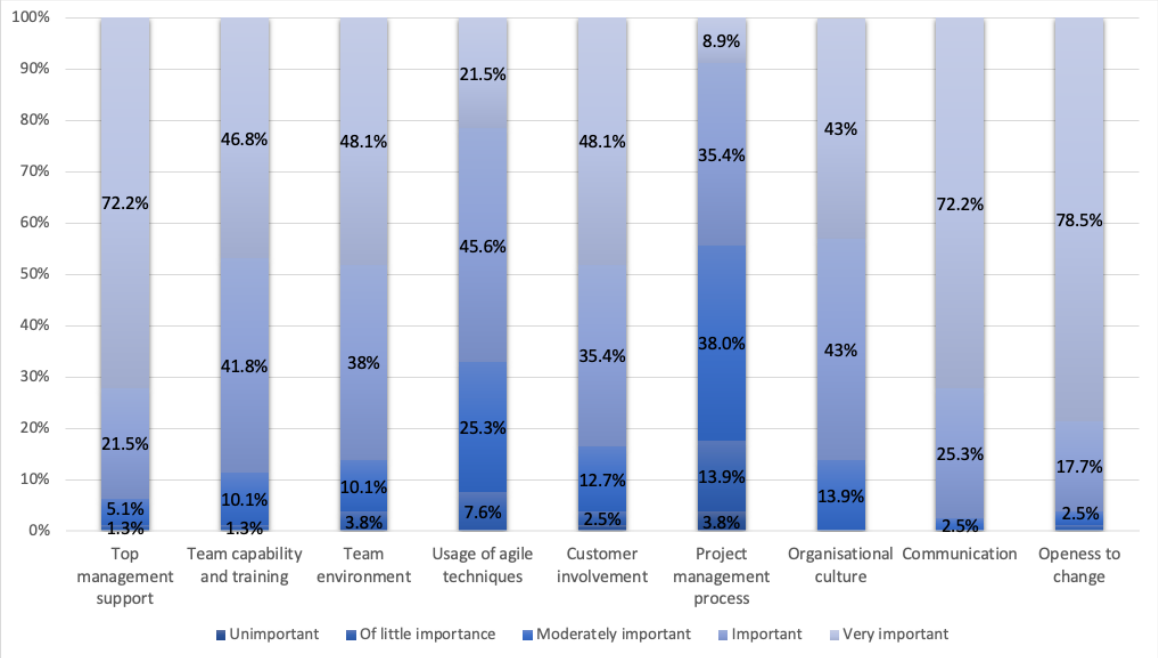
benefits of Agile methodologies. Therefore, it can be inferred that adopting Agile methodologies is perceived to bring benefits to the MSD IT teams.

Further in this study, another question sought to determine the essential factors for the successful implementation of an Agile approach among MSD IT staff. To this end, a Likert scale was employed, where respondents rated the importance of each factor. The aim was to provide insight into the factors that MSD IT employees considered vital during Agile implementation, and consequently, to identify critical areas for attention during the implementation process. To ensure the validity of the results, the factors were sourced from various public studies and theories from the theoretical part of the thesis.

The use of a Likert scale allowed the respondents to express their views on the relative importance of each factor, thereby enabling the researcher to obtain a more nuanced understanding of the participants' perspectives. It is worth noting that the scale also enabled the identification of factors deemed unimportant by the respondents, which could help identify areas where resources could be diverted to more critical areas.

The results of the question identifying the most critical factors to the success of implementing an Agile approach are presented in a Likert graph on the ensuing page *(Graph 5)*, which provides a clear overview of the data. The findings from this study provide valuable insights into the factors that MSD IT staff consider vital for successful Agile implementation, which could be useful for practitioners and researchers in the field.

Graph 5 Share of MSD IT employees evaluating specific factors for successful Agile implementation, n = 79



Source: own processing

The successful implementation of the Agile approach is of paramount importance to the organisation. The results of the data obtained from the question identifying critical factors for a proper Agile implementation indicate that MSD IT staff considers openness to change, top management support, and communication as the most critical factors for the successful implementation of an Agile approach. These findings are consistent with other studies, which most of them have reported that openness to change and top management support are critical to the success of Agile implementation.

The study found that 78.5% of respondents rank openness to change as very important, with no respondents marking it as not important. This finding is consistent with the view that the Agile approach requires a culture of continuous improvement and adaptation, where openness to change is essential. The importance of top management support was also highlighted, with 72.2% of respondents indicating that it was very important. The study's findings support the research from Kim & Lee that emphasises the importance of top management support for successful Agile implementation (Kim & Lee, 2021).

Subsequently, communication is also considered critical, with 72.2% of respondents ranking it as very important. Effective communication is essential for Agile implementation, as it allows for the free flow of information and the development of trust within teams. Also, the State of Agile report suggests that effective communication and collaboration among team members and stakeholders is also a critical factor for success in Agile projects.

Conversely, the study found that the project management process was considered the least important factor, with three respondents marking it as not important, and 13.9% of respondents marking it as of little importance. This finding suggests that MSD IT staff does not probably consider the traditional project management approach as well-suited to Agile implementation. This is consistent with the Agile approach's emphasis on self-organising teams and adaptive planning.

Next, customer involvement was considered essential by a larger proportion of respondents (48.1%), and only one respondent considers it as unimportant. The importance of customer involvement in Agile implementation is consistent with the Agile approach's customer-centric focus. However, the factors project management process and customer involvement were the only factors identified as unimportant. Interestingly, even though only one respondent considers customer involvement as an unimportant factor and two respondents as of little importance, it is still surprising finding because customer Involvement is one of the main features of Agile, so it is an important step during the Agile approach, which is where it differs from other, traditional methods. According to the 15th State of Agile report, the top two factors for Agile success are management support and customer involvement.

Team capability and training are also considered as critical, with 46.8% of respondents indicating it as very important, and 41.8% indicating it as important. The study's findings suggest that the organisation needs to invest in its teams' training and development to ensure successful Agile implementation, which MSD IT does. MSD IT provides its employees with various pieces of

training, such as a product management academy or Scrum Master academy, and with many other insights. Moreover, this finding is consistent with other research that emphasises the importance of team capability in Agile implementation. The State of Agile report highlights that team capability and training are critical factors for Agile success, and this finding is consistent with the results of the MSD IT staff survey. The report indicates that 43% of respondents believe that team capability and training is a critical factor for Agile success, and in the MSD IT staff survey, 88.6% of respondents identify team capability and training as important or very important.

Finally, the team environment is also rated as essential by a significant proportion of respondents, with 48.1% of respondents ranking it as very important, and 38% as important. The study's findings suggest that the organisation should create a supportive team environment to ensure successful Agile implementation. This finding is consistent with the research from Lee & Lee that emphasises the importance of a positive team environment in Agile implementation (Lee & Lee, 2021).

In summary, the study's findings suggest that the organisation seeking to implement the Agile approach should prioritise openness to change, top management support, and communication. MSD IT employees consider these factors as the most critical. The findings also highlight the importance of customer involvement, team capability and training, and a supportive team environment for successful Agile implementation. Additionally, the results of the MSD IT staff survey are consistent with the findings from the State of Agile report, where management support, customer involvement, communication, team capability and training, and team environment are identified as the critical factors for successful Agile implementation. Conversely, the study revealed that the project management process was regarded as the least significant factor, with the usage of Agile techniques being selected as a moderately important factor. In general, the study's findings about critical factors for successful implementation provide valuable insights for organisations seeking to implement the Agile approach and add to the body of knowledge on factors critical to Agile implementation success.

The questionnaire contained another open-ended question that allowed respondents to express their opinions and views on the impacts of Agile development practices on their work. In particular, the following analysed question aimed to explore the respondents' perspectives on the changes and impacts of Agile development practices on their work outcomes. Only those respondents who work at least partially in an Agile way in their teams still answered this question. Thus, a total of 79 MSD IT employees answered the question on the impact of Agile development practices on their work. This question has been formulated with the purpose of obtaining more specific information for the company MSD, which aims to explore individual responses and gain a more comprehensive understanding of the sentiments and perspectives of the respondents regarding the implications of employing an Agile approach.

Based on the responses to the question asking about impact MSD IT staff see the Agile development practices having on the outcomes of their work, in general, the majority of the respondents acknowledged the positive impact of Agile development practices on the outcomes of

their work. Respondents commonly reported that Agile development practices helped them to create a common language, technique, and process to get things done, deliver products faster and more frequently, and improve customer satisfaction. Additionally, respondents highlighted that Agile development practices enabled them to structure their work in shorter iterations, experiment more, and get feedback from stakeholders on a more frequent basis.

Moreover, MSD IT staff emphasised that Agile development practices allowed them to better articulate the desired outcomes and opportunities, prioritise their work more effectively, and collaborate more with their team members. Furthermore, respondents reported that Agile development practices increased the value of their work, provided better transparency and organisation, and improved communication within the team and with stakeholders.

Other important benefits mentioned by many respondents included the ability to adapt to changes quickly, work in smaller iterations, and avoid over-analysing their work. Additionally, Agile development practices increased the level of engagement, enthusiasm, and happiness of the team members, resulting in a more efficient and effective work process.

Interestingly, respondents also mentioned a shift in their way of thinking towards experimentation in short spurts and increased awareness of the broader environment and the impact of their daily work. Additionally, some respondents noted an increased feeling of engagement and happiness among team members, greater collaboration and efficiency, and a better feedback loop across teams.

These findings are consistent with public previous studies that have reported positive impacts of Agile development practices on project outcomes. For instance, the 15th and 16th State of Agile reports revealed that organisations that have implemented Agile development practices experienced increased productivity, faster time-to-market, and higher-quality products. The positive outcomes of Agile development practices can be attributed to its emphasis on collaboration, flexibility, and responsiveness to change.

However, some respondents did mention that the promise of Agile to speed up the delivery of value was not always realised, with more time often needed to adjust to the new way of working. It is recommended that the organisation MSD provide sufficient training and support to their employees from the beginning when implementing Agile development practices to maximise the potential benefits. Respondents also noted that Agile development practices should not be used only as a tool to follow the process but rather to create real value for the customers. Overall, MSD IT employees recognised that Agile development practices have a positive impact on their work outcomes, making it more efficient, effective, and customer oriented.

The following question was designed to identify the reasons why Agile methodologies were not being used in MSD IT teams. Therefore, the question was presented only to those respondents who indicated in the questionnaire survey that they are not working in an Agile way in their teams, which consisted of a total of four respondents. The question was structured as a multiple-choice

question with unlimited answers, and respondents were also given the option to provide their own answers.

The results of the survey show that two out of four respondents stated that they lacked sufficient knowledge about Agile methodologies. These results are consistent with the following question, which results will be analysed below, where respondents, including MSD IT employees working in an Agile way in their teams, also mentioned "lack of skills/experiences with Agile methods" as one of the three most common barriers to adopting Agile practices. The study by Chowdhury (2018) has also identified a lack of knowledge and experience with Agile as a key barrier to adopting Agile methodologies (Chowdhury, 2018). This highlights the importance of providing training and education on Agile methodologies to employees, especially those who are unfamiliar with it.

One respondent stated that he/she does not even know the reason for not using Agile methodologies, which is inconsistent as this knowledge could be expected from the respondents' level of expertise within MSD IT. It may suggest a potential lack of communication and awareness within the organisation about the benefits of Agile methodologies. Much public research has highlighted the importance of creating a culture of communication and collaboration within organisations to foster innovation and creativity related to Agile. This could involve providing regular updates and information about Agile methodologies to employees, as well as encouraging open communication channels for feedback and ideas.

The remaining respondent indicated that their team was working to apply Agile methodologies but had not yet fully mastered the skills required for successful implementation. This suggests again that there may be a need for additional training and support to ensure that all MSD IT teams have the necessary skills and knowledge to implement Agile methodologies effectively. In short, the ongoing training and support can help the organisation successfully adopt and implement Agile methodologies.

In comparison with the State of Agile report, the 16th State of Agile report provides similar insights into the most common reasons why organisations do not adopt or implement Agile methodologies, and that is the inability to change organisational culture (49%), insufficient experience with Agile (43%) and resistance to change (39%). This is also consistent with the analysis of the results of the following question, which highlights these barriers.

Overall, the majority of respondents indicated they lacked sufficient knowledge, experience, and skills about Agile as a reason for not working in an Agile way. These results highlight the need for MSD IT to provide training and education on Agile methodologies to its employees, as well as to create a culture of communication and collaboration to foster innovation and creativity. Ongoing training and support should also be provided to ensure that MSD IT teams have the necessary skills and knowledge to implement Agile methodologies effectively. By addressing these issues, all of the MSD IT teams can reap the benefits of Agile methodologies, including better communication and

collaboration within the team, increased visibility and flexibility, faster time-to-market and many other benefits.

The following question explored what the most significant barriers are to adopting Agile practices in MSD IT staff teams. Like most questions, this question was a multiple-choice type of question; however, there were limited answer choices. Respondents could select up to three answers, three barriers that are the most significant to them during the adoption of Agile practices. The list of responses included the most common barriers that may occur during the adoption of Agile. The answers were created by taking inspiration from published studies and reports as well as by brainstorming the Agile team's own ideas. Anyway, as usual, if a respondent did not find a suitable answer for him/her in the answer choices, the option was to choose "other" and complete the answer in his/her own words. Furthermore, it is possible that respondents did not even encounter any barrier during the implementation of Agile practices in their team; hence, the "none" option was included in the list of options also. It is important to note that unlike the previous questions that were displayed only those respondents using the Agile approach, the following analysed question was shown to all respondents, so all 83 respondents answered this question, whether or not a respondent work in an Agile way in their team.

From the responses provided in question identifying the barriers, it seems that the most significant barriers to adopting Agile practices in the team are related to the culture and mindset of the organisation, as well as the lack of understanding of Agile methods. The most frequently predominantly mentioned barrier is "resistance to change", which was selected by most of the respondents, and that is 55.4%. Resistance to change is one of the most common barriers in any organisational change process, including the adoption of Agile practices. It suggests that some team members or managers may not be comfortable with the idea of changing the way things have always been done. To overcome this obstacle, it is important to have open communication and involve all stakeholders in the change process. Change management strategies can also be used to help people understand why the change is necessary and how it will benefit the team and the organisation.

With a 22.8% decrease, the second most selected barrier to adopting Agile practices is "pervasiveness of traditional development methods", selected by 32.5% of respondents, and subsequently, 31.3% of respondents selected the barrier "lack of skills/experience with Agile methods". The pervasiveness of traditional development methods is closely connected to the most common barrier – "resistance to change". The barrier related to a lack of understanding of Agile principles and practices suggests that team members may not have a clear understanding of what Agile is and how it works. To overcome this barrier, it's important to provide training and education on Agile principles and practices. This can include workshops, seminars, and coaching sessions. It's also important to have ongoing communication and feedback to ensure that team members understand the concepts and how to apply them in their work.

The other common barriers are "inconsistent processes and practices across teams", which is a significant barrier for 20 respondents (24%), and "lack of wider buy-in around Agile", selected

by 19 respondents (22.8%). Inconsistent practices can make it difficult to align project goals, timelines, and priorities, which can ultimately affect the success of Agile implementation. To address this barrier, the organisations can establish clear and consistent Agile processes across teams, and it is also important to establish regular communication channels between teams to foster collaboration and ensure alignment on project goals and priorities.

What is quite surprising is that the findings of this question reveal a disconcerting trend whereby a considerable proportion of the participants (19.3%) identified the lack of management support as a significant obstacle in embracing Agile practices within their respective teams. It suggests that there may be resistance to change from those in positions of authority, which could be hindering the adoption of Agile practices. Therefore, the barrier "lack of management support" may be also closely related to the most common barrier – "resistance to change", as it suggests that managers or leaders may not be fully on board with the adoption of Agile practices. Moreover, without the support of management, it can be difficult to get buy-in from the rest of the team. To overcome this barrier, it's important to communicate the benefits of Agile practices and how they align with the organisation's goals. Demonstrating the success of Agile practices in other organisations can also be helpful.

Another surprising result is that a small but significant number of respondents (8.4%) identified "insufficient training and education" as a barrier to adopting Agile practices, which also indicates that there may be a lack of knowledge and expertise on Agile methodologies within the team. This barrier is in conjunction with the barrier of a lack of knowledge about Agile but focuses specifically on the need for training and education. Without the necessary knowledge and skills, it can be difficult to adopt Agile practices effectively. To overcome this barrier, it's important to provide the necessary training and education for all team members, including managers and leaders. Nevertheless, it is known that MSD IT is trying to work on this barrier and provides various types of pieces of training and academies, such as the Product Owner Academy or Scrum Master Academy, plus individual consultations about what is needed related to Agile for its employees. Additionally, MSD also set up Agile coaches' development teams to help teams implement Agile practices effectively and address any concerns or challenges that arise during the process. Therefore, the obtained result is deemed surprising. Perhaps there is a rationale and not all employees know about sufficient training and education; hence the Agile team could increase awareness of all available Agile training and academies.

On the contrary, problems with using of Agile tools and poor collaboration and knowledge sharing can be considered as the least frequent barriers. Nevertheless, lack of open communication, selected by 8.4%, and poor collaboration and knowledge sharing that makes a barrier for 4 respondents, could be indications that there is a lack of effective communication and collaboration in some teams, hindering the adoption of Agile practices. These barriers suggest that there may be issues with communication and collaboration among team members. Agile practices rely heavily on collaboration and communication, so addressing this barrier is essential. To overcome this barrier, it's important to foster a culture of collaboration and communication. This can include regular

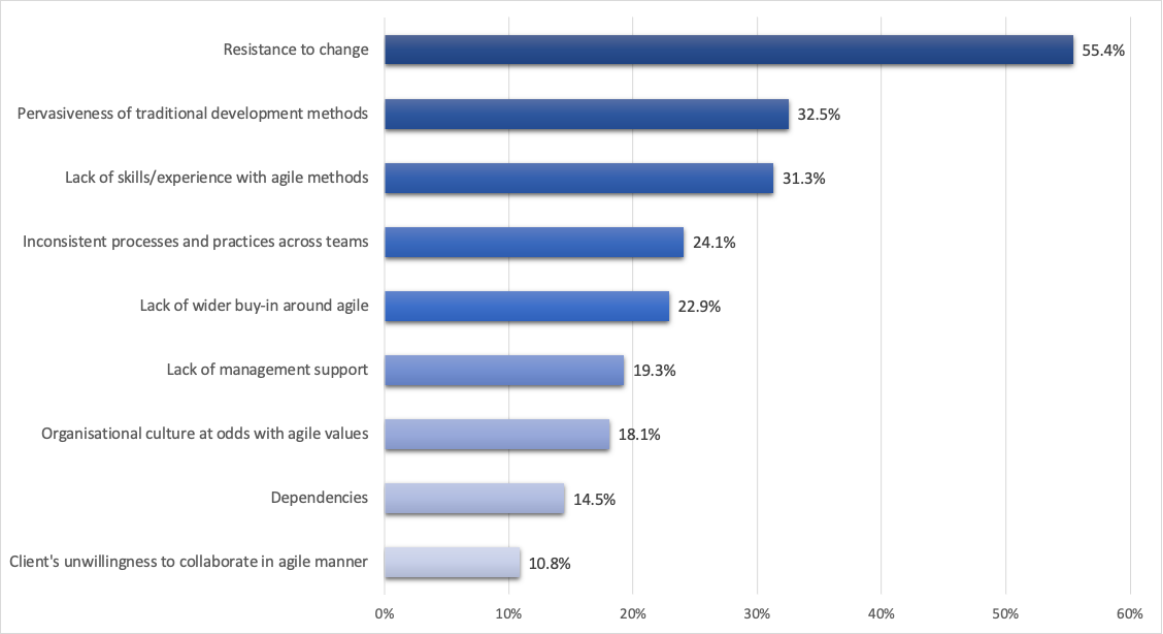
meetings, open communication channels, and tools that facilitate collaboration and knowledge sharing.

A positive result is that, although not many respondents, three respondents chose the option "none", and therefore that they have no significant barriers in the team during the adaptation of Agile practices. Furthermore, a total of 6 respondents also chose the option "other", where they added "changing requirements", "slow and long decision-making process", "business model", or also "not-having a proactive mindset" among the barriers they encountered to adopting Agile practices in their teams. One respondent added multiple barriers to the "other" option, namely lack of time for one project, multitasking and specific environment and procedures in MSD and pharma. One respondent from the MRL IT division added that most teams do not prioritise the work to establish CICD and automated testing; in his/her opinion, if one does not prioritise the work to establish the basics, one will never be able to do quick releases no matter what other Agile practices one attempt.

In conclusion, the outcomes of the inquiry related to the barriers survey question are consistent with most studies and reports that state that the biggest challenges to Agile adoption are organisational culture, resistance to change, and lack of experience with Agile practices. The results exhibit a notable correspondence specifically with the 15th and 16th editions of the State of Agile reports, which recognise resistance to change, inconsistent processes and procedures across teams, and lack of management support as the principal three obstructions to implementing Agile practices in corporate environments. Specifically, the three most significant impediments for MSD IT personnel to embrace Agile practices are “resistance to change,” “pervasiveness of traditional development methods,” and “lack of skills/experience with Agile methods.” Conversely, “problems with using of Agile tools” and “poor collaboration and knowledge sharing” may be regarded as the least significant obstacles. In summary, the survey responses suggest that there are several barriers to adopting Agile practices in the team, but they can be overcome with the right strategies and mindset. Addressing the culture and mindset of the organisation, providing education and training, and fostering collaboration and communication are key to successfully adopting Agile practices. In general, the survey results suggest that in order to overcome these barriers and successfully adopt Agile practices, there needs to be a concerted effort to educate and support team members, foster a culture of collaboration and communication, and address any resistance to change from management.

To present a more accurate representation of the findings from the barrier survey question, the following graph illustrated on the next page *(Graph 6)* depicts the nine most significant barriers, chosen by over 10% of respondents, that MSD IT staff encounter when adopting Agile practices in their teams. The remaining barriers, selected by less than 10% of respondents, are itemised in the table situated on the subsequent page *(Table 8)*.

Graph 6 Share of MSD IT employees considering top 9 barriers to adopting Agile practices as most significant, n = 83



Source: own processing

Table 8 MSD IT employees considering other barriers to adopting Agile practices as most significant, n = 83

Barrier to adopting agile practices	Number of respondents	Share of respondents
Lack of time to adapt agile	8	9.6%
Insufficient training and education	7	8.4%
Unwilling to admit mistakes and learn from delivery failure	7	8.4%
Lack of open communication	7	8.4%
Other	6	7.2%
Poor collaboration and knowledge sharing	4	4.8%
None	3	3.6%
Problems with using of agile tools	2	2.4%

Source: own processing

This study also aims to investigate the factors that contribute to the non-adoption of Agile practices by MSD IT staff. The next paragraphs focus on respondents who do not employ Agile approaches in their teams, despite the potential benefits. Although the number of such respondents is relatively small, a comparison of their responses can provide insight into the barriers to Agile adoption. The non-adoption of Agile practices by team members can be influenced by several factors, including resistance to change, limited resources, lack of communication, fear of failure, lack of training, and lack of support from leadership. A lot of studies have identified several factors that can contribute to the non-adoption of Agile practices. One possible factor is a lack of understanding of the principles and benefits of Agile methodologies. According to research, team members may be resistant to change and may not want to learn new skills or ways of working that could be required with Agile methodologies. A lack of training and resources can also hinder adoption, as can a fear of failure and limited communication channels.

To gain further insights into the barriers to adopting Agile practices, the study compared the responses of the four respondents who do not use Agile practices; their reasons for not using Agile with results from the question, which sought to identify the specific significant barriers faced by respondents who not using Agile. Interestingly, each of the respondents who do not use Agile approaches identified three different barriers as the most significant obstacles to the adoption of Agile practices. The findings further suggest that the respondent who mentioned that he/she does not know the reason why he/she does not work in an Agile way, in terms of the significant barriers to adopting Agile practices, the respondent marked "pervasiveness of traditional development methods", "dependencies" as the most significant barriers, and added "slow and long decision-making process" as the third barrier in his own words. One reason dependencies can be a barrier to adopting an Agile approach is that they can slow down the development process. If one team is dependent on another team to complete their work, any delay in the dependent team's work can cause a bottleneck in the process. Another challenge with dependencies is, for instance, that they can make it challenging to implement Agile practices, such as continuous integration and delivery.

Two of the four respondents not working in an Agile way, who stated that they do not have enough knowledge about Agile as a reason for not working in an Agile way both logically selected "lack of skills/experience with Agile methods" as a significant barrier to adopting Agile, and at the same time, both selected "inconsistent processes and practices across teams" as next significant barrier. This finding suggests that the organisation may need to take a more holistic approach to implementing Agile, ensuring that all teams are aligned with the same Agile practices and principles. This can help to avoid confusion and minimise resistance to change. In addition, only in the third most significant barrier choice they differed; one of the respondents selected "lack of management support" as the third most significant barrier, and the other one chose "organisational culture at odds with Agile values". The organisational culture at odds with Agile values can hinder the organisation's ability to deliver value to customers. Furthermore, it can result in a lack of trust and transparency among team members and between teams and management. Agile methodologies require organisations to be customer-centric and focused on delivering value incrementally, and open communication and transparency to work effectively. Moreover, without proper leadership support, team members may not feel motivated to adopt new practices.

The following contingency table presented on the next page *(Table 9)* illustrates the individual responses of non-working respondents gathered from the question asking for reasons for not using the Agile approach in conjunction with the results obtained from the question identifying barriers. The purpose of including this table is to enhance clarity and facilitate a comprehensive overview of the data.

Table 9 Comparison reasons vs. Barriers to adopting Agile practices of respondents not using Agile, n = 4

Reasons for not using the agile approach x Barriers to adopting agile practices
I don't know.
• Dependencies
• Pervasiveness of traditional development methods
• Slow and long decision making process
We don't have enough knowledge about agile.
• Inconsistent processes and practices across teams
• Lack of management support
• Lack of skills/experience with agile methods
• Organisational culture at odds with agile values
We work to apply and don't completely have the skill application down yet.
• Lack of management support
• Lack of skills/experience with agile methods
• Resistance to change

Source: own processing

For more specific research on respondents not using Agile, another question was used to identify the expectations from their customers, which may have implications for the adoption of Agile practices. The question, which explored the expectations from MSD IT staff customers is compared with the question about IT area, which identified the IT group in which the respondents work; thus, what the expectations in the individual divisions are. In general, the results show that the problem of non-adoption of Agile practices is not limited to one specific area of MSD, and customers in every IT area have various expectations. The results indicate that all respondents who do not use Agile practices reported that their customers expect them to work with them and regularly gather feedback, which suggests that using an Agile approach would be beneficial to these respondents. Additionally, two of the four respondents who do not use Agile practices reported that their customers expect them to be able to change priorities quickly, indicating that their teams should be prepared for change, which could be better facilitated through Agile practices.

It is important to note that some respondents reported multiple expectations from customers, which could pose challenges in meeting all of them. For example, fulfilling customer requirements while also accommodating changes in priorities can be challenging and may require the adoption of Agile practices. This can pose challenges for IT teams that must meet multiple customer expectations simultaneously. The need to fulfil customer requirements while also accommodating changes in priorities can be particularly difficult to address when customer requirements are complex, and priorities are continually shifting. To address this challenge, all of the teams may need to adopt Agile practices, such as iterative development, continuous feedback, and adaptive planning. Agile practices can help teams to respond quickly to changing customer needs and adjust their priorities accordingly.

Finally, the adoption of Agile practices by all of the MSD IT teams can provide several benefits to the organisation in meeting customer expectations. For instance, iterative development can help teams to deliver working software incrementally, allowing them to get feedback from customers early and often. This feedback can then be used to adjust priorities and ensure that the final product meets the customer's expectations. Furthermore, Agile teams can work closely with customers to define and refine requirements throughout the development process. This can help to ensure that the final product meets the customer's needs and expectations while also minimising the risk of misunderstandings or miscommunication.

Overall, this analysis provides valuable insights into the reasons and barriers to Agile adoption in the context of MSD IT staff. These findings suggest that the success of Agile adoption may depend on various factors, such as the level of support from management and the overall organisational culture. It is essential for MSD to address the barriers to Agile adoption identified in this study and promote a culture that values Agility and innovation. In general, the organisation should assess and align its organisational culture with Agile values to successfully adopt and benefit from Agile practices. To improve the organisation's overall Agility, all MSD IT teams should work in an Agile way and try to avoid barriers to adopting Agile practices; at the same time, all of the MSD IT employees should feel like they have management support who is trying to move away from traditional development methods and to lead to consistent processes and practices across teams.

In conclusion, the identification of the factors and barriers to adopting Agile practices is critical for the organisation, and the organisation should try to address them to ensure the successful implementation of Agile methodologies by all MSD IT employees. Adopting Agile practices by all of the MSD IT teams and members can help them to respond quickly to changing customer needs, adjust their priorities accordingly, and manage customer expectations effectively.

A contingency table *(Table 10)*, presented on the following page, has been included to compare and illustrate the responses of all respondents not working in an Agile way. The table facilitates a comparative analysis of the data, aiding in the comprehension of the relationship between the IT areas and expectations from their customers.

Table 10 Comparison IT areas vs. Expectations from customers of respondents not working in an Agile way, n = 4

IT area x Expectations from customers
CTO
▪ They expect us to work with them and regularly gather their feedback.
Digital MMD
▪ They expect us to be able to change priorities quickly. ▪ They expect us to be predictable. ▪ They expect us to meet exact delivery dates. ▪ They expect us to work with them and regularly gather their feedback.
EIT
▪ They expect us to deliver frequently. ▪ They expect us to fulfil their requirements exactly. ▪ They expect us to work with them and regularly gather their feedback.
IT Strategy Realization
▪ They expect us to be able to change priorities quickly. ▪ They expect us to deliver frequently. ▪ They expect us to understand their work and come up with solutions. ▪ They expect us to work with them and regularly gather their feedback.

Source: own processing

Finally, this study aims to provide insights into the Agile practices adopted by MSD IT teams and their effectiveness in different types of work. The last paragraphs of the thesis will focus on a comparative analysis of two questions: one of the main research questions, which explored the most effective Agile practices for MSD IT employees, and the question about type of work MSD IT staff mostly doing. As was previously mentioned the former question sought to explore the respondents' opinions on the most effective Agile practices adopted by MSD IT teams, the latter question aimed to understand the types of work that the respondents are primarily engaged in. Additionally, it is pertinent to reiterate that in relation to the question, which inquired about the most efficacious Agile practices as perceived by the MSD IT employees, respondents were afforded the liberty to designate a maximum of three Agile practices.

By comparing the results of these two questions, this study aims to identify the most effective Agile practices for different types of work within MSD IT teams. It is important to note that the focus will only be on the opinions of respondents who work in an Agile manner, ensuring that the comparison is relevant. Moreover, to the understanding of results being easier, the respondents who respond that they do not know what type of work mostly doing and those who do something else, were removed from the analysis. Thus, the last comparison analysis will be based on the responses of 68 respondents since 8 respondents selected another type of work, like for example Sales and Marketing, Consultancy and Coaching, Data Analytics and Digital Solutions and Validation/Testing, and 3 respondents even do not know what type for work they mostly do. Therefore, this study's comparative analysis will provide valuable insights into the most effective Agile practices for types of work as product, platform, service, and program within MSD IT teams. These types of work an organisation is mostly doing can have an impact on the specific Agile

practices that are most effective. However, at a high level, the core Agile principles and values remain the same regardless of the type of work being done.

The type of work a team is doing (product, platform, service, or program) can impact the effectiveness of Agile practices to some extent. While Agile practices can generally be applied across various types of work, certain practices may prove more effective for particular types of work. For instance, product development teams may benefit from practices such as product backlog and outcome-oriented prioritisation. The Product Owner, responsible for maintaining the product backlog, can work closely with the team to ensure that the appropriate features are being developed and prioritised based on customer needs. For platform development teams, technical Agility practices such as continuous integration and testing may be more advantageous. These practices can help ensure platform stability and reliability and enable swift and efficient changes. Service development teams may find practices such as frequent product delivery and customer need discovery most helpful. These practices can ensure that the service is meeting customer needs and evolving based on feedback. For program development teams, practices such as daily stand-ups and sprint planning can be effective. These practices enable the team to work together effectively towards the program's objectives and maintain alignment.

The present analysis presents a contingency table *(Table 11)* illustrated on the following page. The types of work are arranged in columns, while the rows represent the Agile practices deemed most effective by the respondents. It is worth noting that the current investigation aims to investigate the relationship between these two variables, namely the types of work and the perceived effectiveness of Agile practices. To this end, the contingency table provides a comprehensive overview of the association between the two variables, thereby aiding in the analysis of the data collected from the respondents.

Table 11 Comparison number of respondents considering the Agile practices most effective vs. Type of work, n = 68

Effective agile practices x Type of work	Product	Platform	Service	Program	Sum
Convey information face to face	1	0	4	1	**6**
Customer need discovery	9	1	5	1	**16**
Daily Stand-ups	3	0	7	3	**13**
Examine team's effectiveness (Review)	1	0	6	0	**7**
Feedback loops	6	0	4	1	**11**
Frequent product delivery	1	1	2	1	**5**
Having Product Owner	10	3	4	1	**18**
Having Scrum Master	8	0	10	3	**21**
Kanban board/Scrum board	5	0	4	1	**10**
Outcome oriented prioritisation	5	1	3	0	**9**
Product backlog	8	3	5	1	**17**
Retrospectives	3	1	6	1	**11**
Small self-organised teams	3	2	1	1	**7**
Sprint Planning	5	0	4	0	**9**
Technical agility (XP practices)	0	0	0	0	**0**
Working in iterations	8	2	4	0	**14**
Working together daily	8	0	9	0	**17**
Sum	**84**	**14**	**78**	**15**	**191**

Source: own processing

The results presented in the contingency table provided on the preceding page reveal that out of 31 respondents working primarily on product development, 32.3% considered "having Product Owner" as the most effective Agile practice. This result is consistent with the findings of the 16th State of Agile report, which identified "Product Owner" as the most common role in products Agile teams worldwide. Additionally, 29% of respondents selected "customer need discovery" as the second most effective Agile practice for product development teams. This finding aligns with the Agile Manifesto's value of "customer collaboration over contract negotiation," which emphasises the importance of understanding customer needs in product development. The selected Agile practices are essential for product development teams because they help ensure that the product being developed meets the needs and expectations of the end-user and is delivered within the constraints of the project. By having a Product Owner, the development team has a single point of contact who can provide clear guidance on what needs to be built, and the priority order in which features should be delivered. This helps ensure that the development team is focused on delivering a product that meets the needs of the end-user, while also meeting the project's goals and timeline. Customer need discovery is an essential part of the Agile development process because it allows the product development team to understand the needs and expectations of the end-user. By conducting research and gathering feedback from customers, the development team can gain insights into what features and functionalities the product should have, and how it should be designed to ensure that it is intuitive and user-friendly. This helps ensure that the final product is aligned with the customer's needs and expectations, leading to a higher level of customer satisfaction.

For platform development, the study has a small sample size of only five respondents, with "having Product Owner" and logically "product backlog" identified as the most effective Agile

practices by three out of five respondents. While this result cannot be directly compared with the other studies, it is quite consistent with State of Agile reports which highlight the importance of having clear product requirements and ownership in platform development. Agile methodologies prioritise flexibility and adaptability, and these qualities can be particularly useful in platform development because the needs and expectations of users may evolve over time. Well-defined product requirements that are included in the product backlog serve as a critical tool for enabling the development team to work efficiently and effectively towards achieving the desired outcome. Moreover, the presence of a designated Product Owner, who is the key decision-maker responsible for defining and prioritising product requirements, managing the product backlog, and ensuring that the team remains focused on the most critical tasks, further enhances the efficiency and effectiveness of the development team.

Further, the study found that out of 27 respondents working primarily in service development, "having Scrum Master" was selected as the most effective Agile practice by 37% of respondents. This finding is consistent with the 15th State of Agile report, which identified "Scrum Master" as the most common role in Agile service teams worldwide. Additionally, "working together daily" was selected as the second most effective Agile practice by 33.3% of respondents. This finding aligns with the Agile Manifesto's value of "individuals and interactions over processes and tools," which emphasises the importance of collaboration and communication in Agile teams providing service. It is beneficial that selected Agile practices are the most effective since these practices are critical for service development teams because they promote collaboration, communication, and effective project management. Having a Scrum Master is essential for service development teams because they are responsible for facilitating the Scrum process and ensuring that the team remains focused on the project's goals and objectives. By having a dedicated Scrum Master, service development teams can remain focused on delivering high-quality service and meeting the needs of their customers. Working together daily is an essential aspect of Agile development and is particularly important for service development teams since regular communication and collaboration between team members facilitate the efficient transfer of information and ensure that everyone is aware of the project's progress, including any issues that may arise. This approach helps the team remain Agile and adaptable, responding quickly to changes in project requirements, customer needs, and other variables. In addition, service work is typically characterised by a strong emphasis on customer experience, and the success of service work is often measured in terms of customer satisfaction; thus, the Agile practice “customer need discovery” is also essential for the service type of work.

Finally, the study had only five respondents working on program development. Similar to respondents providing primarily service, "having Scrum Master" and "daily stand-ups" were identified as the most effective Agile practices. Both Agile practices were selected by three out of five respondents. Although this result cannot be compared directly with the State of Agile reports due to the small size, it aligns with the importance of clear communication and coordination in program development. The Scrum Master helps to facilitate the Scrum process, remove any obstacles that the team may face, and ensure that the team remains focused on the program's goals and objectives. By having a dedicated Scrum Master, program development teams can remain

focused on delivering high-quality work and meeting the program's needs. Daily stand-ups are an essential aspect of Agile development and are particularly important for program development teams. These meetings provide an opportunity for team members to share updates on their progress, identify any obstacles they may be facing, and collaborate on solutions to ensure that the program remains on track. Daily stand-ups promote collaboration and communication among team members and help to identify any potential issues or challenges early on, allowing the team to address them promptly.

In conclusion, the study's findings suggest that the effectiveness of Agile practices may vary depending on the type of work a team is primarily engaged in. Nevertheless, the efficacy of Agile practices is dependent on various factors, such as the field of work, the team's objectives and priorities, the challenges and opportunities encountered, and others. Nevertheless, all MSD IT teams should regularly evaluate their Agile practices and make adjustments as necessary to ensure effective value delivery. Besides, the results are consistent with the State of Agile reports' findings, highlighting the importance of having clear Product Ownership, understanding customer needs, and effective communication and collaboration in any type of Agile team. Further research with a larger sample size would be beneficial for the MSD IT organisation to confirm these findings and to identify other potential factors influencing the effectiveness of Agile practices. Notably, the outcomes of the present investigation may furnish a source of inspiration even for other organisations embarking on the implementation of the Agile approach. By drawing upon the comparative analysis, organisations can identify the Agile practices that have been perceived as most effective in various types of work. Such insights can enable organisations to tailor their Agile practices to meet the unique demands of their specific work environment, thus enhancing the overall efficacy of their implementation of Agile practices.

6.3 Summary Overview of the Data Collected

The present case study primarily aimed to investigate the Agile practices commonly used in MSD IT teams and the practices deemed most effective by MSD IT staff. Specifically, the study sought to answer two research questions:

- RQ1: What are the commonly used Agile practices in MSD IT teams?
- RQ2: What Agile practices do MSD IT staff consider most effective?

To answer these questions, data analysis was conducted. Results showed that most employees (83.5%) utilise the Agile practice of product backlog, indicating its widespread use. Furthermore, 81% of the respondents reported having a Product Owner in their teams who is responsible for maximising the value of the product and the work of the development team. The third most used Agile practice is the use of Scrum or Kanban board, utilised by a total of 77.2% of MSD IT employees. These results are consistent with other studies, reports, and articles related to Agile practices. For instance, the 15th and 16th State of Agile reports highlight the widespread use

of product backlog, Scrum and Kanban boards, retrospectives, and sprint planning in Agile development.

Regarding the second research question, the most effective Agile practice, according to MSD IT staff, is having a Scrum Master. This is in line with Scrum.org, which states that the role of a certified Scrum Master plays a critical role in ensuring the scrum team follows the scrum framework properly and removing any impediments that could hinder the team's progress. Other commonly selected Agile practices as one of the most effective are "having a Product Owner", "working together daily", and "product backlog", which were deemed equally effective.

In summary, it can be posited that while "product backlog" represents the most commonly implemented Agile practice, the IT personnel at MSD maintain that "having Scrum Master" is the most impactful practice in terms of effectiveness. It is important to note that different teams may have different needs and priorities that require different approaches to Agile development. Nevertheless, the present study sheds light on the Agile practices most commonly used and considered as the most effective in MSD IT teams.

The following section provides a summary overview of the data collected from the other research questions. The first open question in the questionnaire sought an overall view of the Agile processes in the organisation. The most common pattern among respondents is that MSD is still in the process of adopting and implementing Agile methodologies. Results indicate that while some teams have made progress and are using Agile effectively, there are still many teams and divisions that are in the early stages of implementation, and there is a varying degree of understanding and usage across the company.

The next research question explored the extent to which respondents work in an Agile way, which influenced the display of other questions in the questionnaire. Results indicate that the majority of respondents (55%) work in an Agile way, while other respondents (40%) reported using specific parts of the Agile approach. The rest minority of only 4 respondents do not work in an Agile way at all.

Regarding the Agile methods used by Agile MSD IT staff, Scrum was predominantly reported as the most used Agile method (by 83.5% of respondents), while Kanban is the second most commonly used Agile method, the Lean approach ranked third with a quite big decrease. When it comes to the Agile tools, the most widely used Agility-supporting tool is the effective messaging tool MS Teams (employed by 86.1%) of respondents. The second most commonly utilised Agile tool by MSD IT teams is JIRA, followed by MURAL.

In terms of the benefits of Agile methodologies, better collaboration within the team is considered the most significant benefit by the majority of employees. Further, improved communication within the team is also seen as an excellent benefit by many respondents. Conversely, increased speed of delivery, faster process of work and better predictability were the only ones rated as poor benefits of Agile. Related to the critical factors for the successful adoption

of Agile practices for the organisation's staff, MSD IT teams consider openness to change, top management support, and communication as the most critical factors for the successful implementation of an Agile approach.

When the respondents described the outcomes of the Agile approach on their work in their own words, the majority of respondents acknowledged the positive impact of Agile development practices on the outcomes of their projects, such as helping them to create a common language, technique, and process to get things done, deliver products faster and more frequently, and improve customer satisfaction.

For non-Agile respondents, a special question for them indicated that a lack of sufficient knowledge, experience, and skills about Agile was indicated as the primary reason for not working in an Agile way. In terms of the three most significant barriers for all MSD IT employees to embrace Agile practices are "resistance to change," "pervasiveness of traditional development methods," and "lack of skills/experience with Agile methods." Conversely, "problems with using of Agile tools" and "poor collaboration and knowledge sharing" may be regarded as the least significant obstacles.

Regarding customer expectations, most customers expect teams to understand their work and come up with solutions. Secondly, many respondents also marked that customers expect teams to deliver frequently. As the third most common expectation from customers, teams are expected to be able to change priorities quickly.

Finally, the demographic question sought the IT areas respondents are part of to ensure that all respondents are part of the sample; thus, responses of the non-IT respondents were removed. The important is that the responses were collected from respondents in each MSD IT division, indicating a balanced distribution of respondents across IT areas. In terms of the types of work MSD IT teams mostly do, most of the MSD IT teams are product development teams, with the second most common type being service.

Furthermore, during the process of data analysis, a number of comparisons were undertaken. Notably, an examination of the number of respondents using Agile practices and the proportion of those respondents who considered these practices effective yielded interesting findings. For instance, the Agile practice of "product backlog" was found to be the most commonly used technique, utilised by more than 80% of respondents. However, it was noted that only a quarter of respondents considered this practice to be one of the most effective. Similarly, the Agile practice of "having a Product Owner" was used by only two fewer respondents than the number using product backlog. Nevertheless, the proportion of respondents considering this practice to be effective was the same as for the product backlog. Conversely, it was found that even though a relatively small proportion of IT MSD employees work together on a daily basis, compared to other Agile practices, this technique is still considered to be one of the three most effective.

An analysis of the reasons why some respondents did not use Agile practices, in comparison to the barriers to adoption of Agile practices for respondents who did not work in an Agile manner,

revealed that respondents who lacked knowledge, experience, and skills about Agile methods cited "lack of skills/experience with Agile methods" and "inconsistent processes and procedures across teams" as the main barriers to adoption. Additionally, when comparing the areas of non-use for IT respondents with their customers' expectations, all non-Agile respondents indicated that their customers expected collaboration and regular feedback, suggesting that the use of Agile methodologies could be beneficial to these respondents.

Further, the study indicated that the problem of non-adoption of Agile practices was not limited to any particular area of MSD, and that customers in each IT area had different expectations. Finally, an analysis of whether the choice of the most effective Agile practices was related to the type of work respondents carried out revealed that this was only a small determinant. The study found that the most effective Agile practice for product development teams was "having a Product Owner," as well as for teams primarily engaged in platform work considered "having a Product Owner" and "product backlog" to be the most effective practices. For development teams providing services, "having a Scrum Master" was found to be the most effective practice, the same as for teams focused on software development considered "having a Scrum Master" and "daily stand-ups" to be the most effective practices. Overall, the type of work was found to be a minor factor in determining the most effective Agile practices, with most teams considering the same Agile practices to be effective.

6.4 Proposals and Recommendations

The study primarily highlights the need for the organisation to provide training and support to its employees in Agile methodologies to bridge the knowledge and skill gap, as reported by many respondents.

Firstly, it is imperative to encourage more MSD IT employees to start using Agile practices to reap its benefits fully because while 40% of the respondents indicated that they use some aspects of Agile methodology, it is not enough to qualify them as completely Agile. It was found that only 55% of MSD IT employees implement and work fully in an Agile way. It means that MSD seems to be striving to be more Agile, and there is still work to be done to integrate Agile methodologies across the organisation. Further research should be conducted to explore if all MSD IT teams work in the proper way. Moreover, subsequent research would be useful to explore the factors underlying the use of Agile methods and frameworks by MSD IT employees. For instance, it is surprising to note that some MSD IT teams openly use SAFe, despite its official discouragement, and it would be essential to explore the factors underlying this practice.

Based on the findings of this study, it was determined that in the context of Agile practices, a large percentage (84%) of employees reported having a product backlog. Furthermore, the majority of respondents (81%) acknowledged the presence of a designated Product Owner in their teams. However, the coexistence of a product backlog without a designated Product Owner suggests a gap in the effective implementation of Agile methodologies in some teams. It may lead

to confusion and misalignment within the team, affecting the quality of the product being developed. Hence, it is crucial to ensure that every team has a Product Owner, Scrum Master, or Agile coach, where needed, and that every one of them is skilled. Having a skilled Scrum Master is crucial for the success of Scrum implementation in teams. It is surprising to note that more respondents reported having a Product Owner than a Scrum Master, possibly due to a lack of understanding of the roles and responsibilities of a Scrum Master in Agile practices. Furthermore, while the study reports that the most effective Agile practice is having a Scrum Master, it would be interesting to explore why this practice is considered the most effective and how it contributes to the success of the teams. Future studies could investigate the specific behaviours and actions of Scrum Masters that contribute to their effectiveness. Similarly, the benefits of product backlog may not be fully understood, leading to underutilisation by some teams; thus, clarifying the role of the Product Owner or improving communication with the rest of the team may help in addressing this issue. It is recommended to clarify Agile roles and apply them in as many teams as possible to improve the effectiveness of the Agile approach.

To be fully Agile, it is recommended that MSD IT teams implement more daily meetings, convey information face-to-face, and hold sprint reviews to examine team's effectiveness. These vital aspects of Agile practices should occur more frequently to improve team communication, collaboration, and productivity. Collaboration, both virtually and in person, is a crucial aspect of Agile practices, and teams should use tools and practices that support collaboration and communication. Face-to-face communication is the most effective way of conveying information within a development team that facilitates quick feedback, collaboration, problem-solving, and a shared understanding of the project's goals. Additionally, if Agile methods are used, the team should always consist of small self-organised teams and involve the customer as much as possible in the process to find out as much information about the customer's needs and requirements. This helps in developing a better product that meets the customer's expectations.

In terms of barriers, the most significant barriers to adopting Agile practices in the MSD IT teams are related to the culture and mindset of the organisation, as well as mentioned the lack of understanding of Agile methods. To overcome these obstacles, the organisation should try to avoid resistance to change, which was also selected by most respondents as a common barrier. Resistance to change is one of the most common barriers in any organisational change process, including the adoption of Agile practices. To overcome this obstacle, it is important to have open communication and involve all stakeholders in the change process. Moreover, the second most selected barrier to adopting Agile practices is the pervasiveness of traditional development methods; thus, the recommendation is to apply change management strategies that can help people understand why the change is necessary and how it will benefit the team and the organisation. Another barrier related to a lack of understanding of Agile principles and practices suggests that team members may not have a clear understanding of what Agile is and how it works. To overcome this barrier, it's important to provide training and education on Agile principles and practices, including workshops, seminars, and coaching sessions. It's also important to have ongoing communication and feedback to ensure that team members understand the concepts and how to apply them in their work.

Furthermore, inconsistent practices can also be a common barrier, making it difficult to align project goals, timelines, and priorities, ultimately affecting the success of Agile implementation. The organisation MSD should establish clear and consistent Agile processes across teams, and it is also important to establish regular communication channels between teams to foster collaboration and ensure alignment on project goals and priorities. Next, lack of management support is also a significant obstacle in embracing Agile practices within their respective teams, and it should not happen. It suggests that there may be resistance to change from those in positions of authority, hindering the adoption of Agile practices. Therefore, the barrier related to the lack of management support may be closely related to the most common barrier, which is resistance to change, as it suggests that managers or leaders may not be fully on board with the adoption of Agile practices, and that is necessary to solve. Finally, it would also be helpful to examine challenges or barriers that MSD IT staff have experienced in more detail with implementing Agile practices, as this could inform future improvements or modifications to the practices.

Therefore, the study's findings suggest that the organisation seeking to implement the Agile approach should prioritise openness to change, top management support, and communication. The organisation should focus mainly on the importance of customer involvement, team capability and training, and a supportive team environment for successful Agile implementation. Some respondents did mention that the promise of Agile to speed up the delivery of value was not always realised, with more time often needed to adjust to the new way of working. Therefore, it is recommended again that MSD provides sufficient training and support to their employees from the beginning when implementing Agile development practices to maximise the potential benefits. Agile development practices should not only be used as a tool to follow the process but to create real value for the customers.

Next, prospective studies could explore the effectiveness of different Agile tools and how they contribute to the success of various types of development projects. Selecting the right tools for MSD IT teams' specific needs is important, and MSD IT teams should consider factors such as team size, project complexity, and desired level of collaboration when selecting Agile tools. MSD IT teams were found to be using a range of Agile tools, including some non-supported ones, depending on their specific needs and preferences. The proposal is to make clear rules within the company in terms of supported Agile methods and tools. A further suggestion is to investigate why benefits such as "Increased speed of delivery," "Faster process of work," "Better predictability," or "Improved team collaboration" may not always be realised, despite the use of Agile tools and to identify the factors that may affect the effectiveness of Agile implementation. Furthermore, it is highly recommended that MSD IT teams regularly evaluate their Agile practices and processes to identify areas for improvement and ensure continuous improvement. This could include conducting regular retrospectives to reflect on what has worked well and what could be improved in each project iteration.

Primarily, the case study reveals that a predominant issue persists among the respondents, whereby a considerable proportion of them exhibit inadequate comprehension regarding Agile methodology. Based on the findings, it can be inferred that a dearth of knowledge concerning Agile

presents a significant challenge within the context of the study since many respondents cited a lack of skills or experience with Agile methods as one of the three most common barriers to adopting Agile practices. This also highlights the importance of providing effective training and education on Agile methodologies to employees, particularly those who are unfamiliar with it. The organisation should consider providing ongoing training and support to ensure that all MSD IT teams have the necessary skills and knowledge to implement Agile methodologies effectively.

The implementation of training programs for employees represents a productive measure to address the knowledge and experience gaps related to Agile practices within MSD IT. As several MSD IT staff members have reported a lack of familiarity with Agile, it is suggested that training could serve as a means of increasing interest in Agile and reducing barriers to its adoption. Despite the provision of various training options by MSD IT, it is clear that a change is needed to make Agile training more effective.

To address this issue, MSD IT could offer in-person training programs at their offices, in addition to the online training currently available. It has been shown that in-person training can increase interest and attentiveness compared to online platforms. The proposed training sessions could run for 2-3 days and be tailored to specific teams and roles within the organisation. The cost of hiring an external Agile coach is estimated by industry average to be around 100-150 USD per hour. The price of an Agile coach can vary depending on many factors such as country, region, coach experience and contract length. However, savings could be made by using internal staff and avoiding the rental of training space. Overall, it is estimated that the cost of providing training for instance for 20 employees over two days would be approximately 2,000 USD, with 1,500 USD allocated for the external coach and 500 USD for refreshments, materials and other variable costs.

Training could cover a range of topics suggested by the results of the case study, including OKRs as a tool to elucidate and facilitate the establishment of quantifiable goals within teams or change management strategies to address one of the most significant barriers to adopting Agile, namely resistance to change. The training could also focus on the evolution of the product model, imparting practical insights into the application of product models and hands-on product knowledge. Furthermore, the training could potentially address novel topics such as funding and financing, thus broadening the scope of the training program. Training sessions should also clarify the different Agile roles and their responsibilities, as not all teams have a dedicated Scrum Master or Product Owner. The selection and use of Agile methods and tools could be supported by clear rules and guidance from the organisation, as some respondents have reported using unsupported methods and tools. Finally, training could include preparation for official Agile certifications, which could increase interest and awareness of Agile practices among MSD IT staff and pay for the associated courses, which typically cost around 250 USD per person. This would incentivise employees to engage in the training and demonstrate their proficiency in Agile methodology.

In conclusion, addressing the lack of knowledge and skills of MSD IT staff in Agile methodology is crucial for the successful adoption of Agile practices within the organisation. The proposed recommendations, which include offering in-person training, customised training for

individual teams and roles, and training on specific topics, as well as providing preparation for Agile certifications, offer a comprehensive approach to bridge the skills gap and improve Agile adoption.

Overall, the study highlights the importance of providing sufficient training and support for all employees to overcome common barriers to adopting Agile practices. Open communication, involving all stakeholders in the change process, and establishing clear and consistent Agile processes across teams are also essential for successful Agile implementation. The organisation MSD should continuously evaluate their Agile practices and processes to ensure that they are delivering value to customers effectively. In general, the organisation should focus more on employees' points of view related to Agile and explore the MSD IT teams and their members in more detail, preferably in a non-anonymous way, and then especially help and support those employees who want it or need it. The results of this study indicate that the implementation of Agile methodologies may be advantageous in meeting customers' expectations for regular feedback and collaboration.

Conclusion

The area of interest for this thesis was Agile project management, explored in more detail from the perspective of MSD's IT staff within the case study. Agile project management and product development have been a trend in recent years. This way of thinking, complemented by methods, practices and techniques, has gained its popularity due to its ability to adapt to changing environments caused by market changes, new technologies, business models or by gradual learning and clarification of mutual misunderstandings. Agile is most commonly used in software development, but is also being adopted in other areas such as IT operations, marketing, and HR. Although there is a lot of emphasis on the implementation of Agile methods in many companies nowadays, it has been confirmed that Agile is still considered a modern method that many people still lack knowledge and experience about. Many companies are attempting to apply Agile principles even deeper than just to their projects and are undergoing what is called an Agile transformation of the entire company, which is what MSD IT is trying to do.

The thematic focus of the thesis is on Agile project management and its essence. In the working environment, there is a frequent inflection of this term and as a result, it is possible to encounter disrespect, denigration of this method and the incorrect application of Agile methods and principles, which was one of the reasons why I chose Agile project management as the topic of my thesis. There is a lot of misunderstanding and superstition around Agile, that it is chaos without plans and management, or that through Agile we deliver many times more content in the same conditions of many competing priorities and teams spread across departments. These are untruths stemming from a misunderstanding of basic Agile principles.

Another reason for choosing this topic focus was the aforementioned fact, and that is the unfamiliarity and still newness of the Agile approach. In most cases, people do not have a concrete idea about the Agile interactive project management approach and its benefits or often have a false negative reputation about the Agile approach. Agile thinking is also often confused with Scrum, one of the Agile process frameworks; hence, many people mistakenly think that Agile equals Scrum. With such a misunderstanding, the adoption of Agile can then be purely mechanical, at the process or organisational structure level, without a deeper change in the mindset of the people involved and, most importantly, without value to the company and its customers. One of the main reasons for choosing this topic is also my own interest in this topic and personal development within Agile obtained from my experience with Agile as part of my internship at MSD IT.

The subject of this thesis is an attempt to explain the definition, nature, importance, benefits, and process of Agile in a comprehensible way, as it is one of the major influences on the use of Agile in companies. The case study provides valuable insights into the use and effectiveness of Agile methodologies in MSD IT and could serve as a basis for further research within MSD IT and the development of best practices in this field.

The intent and objective of the thesis then was to analyse the data obtained from the selected sample of respondents, MSD IT staff to learn about their perspective and opinion about the Agile process within MSD IT in their teams. The sources of information were literature, articles, and studies, acquired expertise and experience from MSD IT Agile coaches and qualitative-quantitative research. The methods used in this thesis were based on the above-mentioned dispositions and parameters.

The thesis started with the experience in the topic and the identification of the current situation in MSD related to Agile. This was followed by the shaping of the thesis in all its stages - preparation, elaboration, writing and evaluation. The whole process ended with the evaluation, summary and comparison of the results, followed by suggestions and recommendations for MSD. The writing of the thesis involved numerous modifications to the originally planned concept, but it was always a constructive improvement and, on the whole, a simplification. Firstly, the text was written, which was then supplemented with graphical elements and the corresponding appendices.

The present study culminates in a comprehensive explication and depiction of the Agile process, along with an in-depth analysis of the current state of the adoption of Agile practices in MSD IT. Consequently, the research questions posed at the outset of the study have been successfully addressed. This investigation, which has centred on the particular context of MSD IT teams, offers valuable insights into the practical implementation of Agile methodologies in real-world scenarios.

The present thesis holds potential for practical application within MSD and other organisations interested in adopting the Agile approach. Additionally, it can serve as a valuable resource for IT professionals, educators, project managers, and those involved in employee development in Agile. The extensive literature review and case study presented provide useful insights, particularly for individuals who lack familiarity with Agile methodologies. These resources may also serve as a source of inspiration for organisations not currently utilising Agile practices. However, it is important to acknowledge that the study's focus on MSD IT teams may limit the generalizability of the findings. Different teams may have varying needs and priorities, requiring customized approaches to Agile development. While the results of this study are informative, they should not be considered universally applicable.

The conclusions drawn from this thesis hold great potential benefits, not only for MSD but also for MSD IT teams, their members, and Agile coaches. This study thus contributes significantly to the advancement of knowledge in the field of Agile project management and offers practical implications for organisations seeking to improve their Agile practices. As such, the findings of this research may also be of value to other organisations, their employees, and those involved in Agile coaching and practice. The COVID-19 pandemic has propelled the adoption of Agile methods and underscored the significance of Agile principles such as adaptability and collaboration; thus, many organisations are trying to implement an Agile approach nowadays. As many organisations strive to embrace an Agile approach, the results of this thesis can serve as a foundation and inspiration for their decision-making. Finally, the aims and objectives of this thesis have been met.

List of the Bibliographic References

Literature resources:

1. *A guide to the Project Management Body of Knowledge* (2017) Newtown Square, PA: Project Management Institute, Inc.
2. BARTOŠOVÁ, H., PONIKELSKÝ, P., BARTOŠ, J. (2012) *Projektový management*. Praha: Vysoká škola regionálního rozvoje. ISBN 978-80-87174-13-5.
3. BUCHALCEVOVÁ, A. (2005) *Metodiky vývoje a údržby informačních systémů: kategorizace, agilní metodiky, vzory pro návrh metodiky*. Praha: Grada. ISBN 80- 247-1075-7.
4. CLUSTER, K. (2019) *Agile Project Management: Learn How to Manage a Project with Agile Methods, Scrum, Kanban and Extreme Programming*. Independently Published. ISBN 9781710291698.
5. COLE, R. (2021) *Agile Now: Your quick start introduction to agile*. Harlow: Pearson Business. ISBN 9781292232027.
6. DOLEŽAL, J., MÁCHAL, P., LACKO, B. (2012) *Projektový management podle IPMA. 2., aktualiz. a dopl. vyd.* Praha: Grada. ISBN 978-80-247-4275-5.
7. HIGHSMITH, J. A. (2013) *Agile Project Management: creating innovative products. Second edition.* Boston: Addison-Wesley Professional. ISBN 978-0-321-65839-5.
8. KERZNER, H. (2017) *Project management: a systems approach to planning, scheduling, and controlling*. Twelfth edition. Hoboken: Wiley. ISBN 978-1-119-16535-4.
9. MORAN, A. (2015) *Managing Agile – Strategy, Implementation, Organisation and People*. Springer. ISBN 978-3-319-16261-4.
10. PREUßIG, J. (2018) *Agiles Projektmanagement: Agilität und Scrum im klassischen Projektumfeld*, Haufe-Lexware, Stuttgart. ISBN 978-3-648-10589-4.
11. SCHWALBE, K. (2011) *Řízení projektů v IT: kompletní průvodce*. Brno: Computer Press. ISBN 978-80-251-2882-4.
12. SUTHERLAND, J.V. (2019) *Scrum: The art of doing twice the work in half the time*. London: Random House.
13. SVOZILOVÁ, A. (2011) *Projektový management. 2. akt. a doplněné vydání*. Praha: Grada. ISBN 978-80-247-3611-2.
14. ŠOCHOVÁ, Z. & KUNCE, E. (2019) *Agilní metody řízení projektů. 2. Vydání*. Brno: Computer Press. ISBN 978-80-251-4961-4.
15. WATT, A. (2014) *Project management*. BCcampus, [online]. Available at: https://opentextbc.ca/projectmanagement/.
16. WOMACK, J. P. & JONES D. T. (2003) *Lean thinking: banish waste and create wealth in your corporation*. Free Press: New York. ISBN 0-7432-4927-5.
17. WYSOCKI, R.K. (2019) *Effective project management: Traditional, Agile, extreme, hybrid*. Indianapolis, IN: John Wiley & Sons, Inc. ISBN 978-1-119-56280-1.

Articles:

18. Ahmad, M. O., Dennehy, D., Conboy, K., Oivo, M. (2018) “Kanban in software engineering: A systematic mapping study,” *Journal of Systems and Software*, 137, p. 96–113. Available at: https://doi.org/10.1016/j.jss.2017.11.045.
19. Aldahmash, A., Gravell, A.M., Howard, Y. (2017) “A review on the critical success factors of Agile Software Development,” *Communications in Computer and Information Science*, p. 504–512. Available at: https://doi.org/10.1007/978-3-319-64218-5_41.
20. Andrei, B., et al. (2019) “A Study on Using Waterfall and Agile Methods in Software Project Management,” *Journal of Information Systems & Operations Management*, p. 125-135. Available at: https://search.proquest.com/docview/2237828314?accountid=159230.
21. Asanur, Y. (2021) “Agile Triangle vs Traditional Iron Triangle,” [Post]. LinkedIn. Retrieved November 2, 2022. Available at: https://www.linkedin.com/pulse/agile-triangle-vs-traditional-iron-yusuf-adanur/.
22. Beemaraj, R.K., Theni, T. (2018) “Six Sigma concept and dmaic implementation,” *Int. J. Bus. Manage*. Res. 3, p. 111–114.
23. Birgün, S. & Çerkezoğlu, B. T. (2019) “A systematic approach for improving the software management process,” *International Journal of Innovation and Technology Management*, 16, 4: 1–28.
24. Bonome, L., Costa, M., Godinho, M., Fredendall, L. D., José, F., Paredes, G. (2018) “Lean, six sigma and lean six sigma in the food industry: a systematic literature review,” *Trends Food Sci. Technol.*, 82, p. 122–133.
25. Chow, T., Cao, D., B. (2008) “A survey study of critical Access factors in agile software projects,” *The Journal of Systems and Software*, No 81, p. 961–971. ISSN 0164-1212.
26. Chowdhury, S. (2018) “Factors influencing the adoption of agile methodologies in software development: A systematic literature review,” *Journal of Software Engineering and Applications*, 11(4), 155-182.
27. DeRue, D. S., Barnes, C., M., Morgeson, F., P. (2010) “Understanding the motivational contingencies of team leadership,” *Small Group Research*, 41 (5), p. 621–651. Available at: https://doi.org/10.1177/1046496410373808.
28. Dixit, R., Bhushan, B. (2019) “Scrum: An agile software development process and metrics,” *Journal on Today's Ideas - Tomorrow's Technologies*, 7(1). Available at: https://doi.org/10.15415/jotitt.2019.71005.
29. Emelyanova, T. et al. (2020) “Teamwork management in the field of software development,” *Proceedings of the Ecological-Socio-Economic Systems: Models of Competition and Cooperation (ESES 2019)* [Preprint]. Available at: https://doi.org/10.2991/assehr.k.200113.102.
30. Forsberg, P., Lindberg, J., Penzenstadler, B. (2020) “Agile software development benefits and limitations: A review of empirical studies,” *IEEE Transactions on Software Engineering*, 46(3), p. 289-307.
31. Henao, R., Sarache, W., Gómez, I. (2019) “Lean manufacturing and sustainable performance: trends and future challenges,” Journal of Cleaner Production, 208, p. 99–116. Available at: https://doi.org./10.1016/j.jclepro.2018.10.116.

32. Ikumapayi, O. M., Akinlabi, E. T., Mwema, F. M., Ogbonna, O.S. (2020) “Six Sigma versus lean manufacturing,” *Elsevier Ltd.,* p. 3275–3281.
33. Jansen, S., Brinkkemper, S., Visser, M. (2020) “Agile practices, transparency and communication: A systematic literature review,” *Information and Software Technology*, 120, 106228.
34. Kaswan, M.S. & Rathi, R. (2019) “Analysis and modelling the enablers of Green Lean Six Sigma implementation using Interpretive Structural Modeling,” *J. Clean. Prod.* 231*,* p. 1182–1191.
35. Kim, S., & Lee, S. (2021) “Top management support and success factors for agile project management,” *Technological and Economic Development of Economy*, 27(3), p. 689–703. doi: 10.3846/tede.2021.14172
36. Koi-Akrofi, G.Y., Akrofi, J.K., Akwetey, M. H. (2019) “Understanding the characteristics, benefits and challenges of Agile IT project management: A Literature Based Perspective,” *International Journal of Software Engineering & Applications*, 10(5), p. 25–44. Available at: https://doi.org/10.5121/ijsea.2019.10502.
37. Kumar, R., Maheshwary, P., Malche, T. (2019) “Inside agile family software development methodologies,” *International Journal of Computer Sciences and Engineering*, 7(6), p. 650–660. Available at: https://doi.org/10.26438/ijcse/v7i6.650660.
38. Laanti, M. (2014) “Characteristics and principles of scaled agile,” *Lecture Notes in Business Information Processing*, p. 9–20. Available at: https://doi.org/10.1007/978-3-319-14358-3_2.
39. Lee, S., & Lee, H. (2021) “Factors influencing agile software development project success in Korea,” *Journal of Software Engineering and Applications*, 14(2), p. 48-57. doi: 10.4236/jsea.2021.142004.
40. Lei, H., Ganjeizadeh, F., Jayachandran, P. K., Ozcan, P. (2017) “A statistical analysis of the effects of Scrum and Kanban on software development projects,” *Robotics and Computer-Integrated Manufacturing*, 43, p. 59–67. doi: 10.1016/j.rcim.2015.12.001.
41. Meredith, J. & Zwikael, O. (2019) "When is a Project Successful?," *IEEE Engineering Management Review*, 47(3), p. 127–134. doi: 10.1109/EMR.2019.2928961.
42. Mihajlović Milićević, J. et al. (2019) “Scrum agile framework in E-business Project Management: An approach to teaching scrum,” *European Project Management Journal*, 9(1), p. 52–60. Available at: https://doi.org/10.18485/epmj.2019.9.1.7.
43. Petrtyl, J., Skalický, J., Vacek, J. (2012) “Agilní projektový management,” *Západočeská univerzita v Plzni*. Available at: https://adoc.pub/queue/agilni-projektovy-management.html.
44. Pika, A., Van Den Berg, E., Stol, K. J. (2020) “Agile software development practices and their impact on project success: A systematic literature review,” *Information and Software Technology*, 121, 106285.
45. Popa, O., Michele, C., Fagarasan, C., Pisla, A. (2021) “Leadership approach towards Agile, Waterfall and Iterative implementation of the software development products,” *IOP Publishing Ltd, IOP Conf. Ser.: Mater. Sci. Eng.*, 1169 012017. doi: 10.1088/1757-899X/1169/1/012017.
46. Radujković, M., & Sjekavica, M. (2017) “Project Management Success Factors,” *Procedia Engineering, 196*, p. 607-615. Available at: https://doi.org/10.1016/j.proeng.2017.08.048.

47. Santos, P.de & de Carvalho, M.M. (2021) “Exploring the challenges and benefits for scaling agile project management to large projects: A Review,” *Requirements Engineering*, 27(1), p. 117–134. Available at: https://doi.org/10.1007/s00766-021-00363-3.
48. Shrivastava, A. et al. (2021) “A systematic review on Extreme Programming,” *Journal of Physics: Conference Series*, 1969(1), p. 012046. Available at: https://doi.org/10.1088/1742-6596/1969/1/012046.
49. Siegel, R., Antony, J., Garza-reyes, J. A., Cherra, A., Lameijer, B. (2019) “Integrated green lean approach and sustainability for SMEs: from literature review to a conceptual framework,” *J. Clean. Prod.* 240, p. 118–205.
50. Stray, V., Memon, B., Paruch, L. (2020) “A systematic literature review on agile coaching and the role of the agile coach,” *Product-Focused Software Process Improvement*, p. 3–19. Available at: https://doi.org/10.1007/978-3-030-64148-1_1.
51. Stray, V., Tkalich, A., Moe, N. B. (2021) “The agile coach role: Coaching for agile performance impact,” *Proceedings of the Annual Hawaii International Conference on System Sciences* [Preprint]. Available at: https://doi.org/10.24251/hicss.2021.817.
52. Sutherland, J. & Schwaber, K. (2017) “The scrum guide: the definitive guide to scrum: the rules of the game,” *ScrumGuides.org*. Available at: http://scrumguides.org/.
53. Tavares, B.G., da Silva, C.E. and de Souza, A.D. (2019) “Risk management analysis in scrum software projects,” *International Transactions in Operational Research*, 26(5), p. 1884–1905. Available at: https://doi.org/10.1111/itor.12401.
54. Thesing, T., Feldmann, C., Burchardt, M. (2021) “Agile versus Waterfall Project Management: Decision Model for Selecting the Appropriate Approach to a Project,” *Procedia Computer Science*. p. 746–756. Available at: https://doi.org/10.1016/j.procs.2021.01.227.
55. Yordanova, S., Toshkov, K. (2019) "An Agile Methodology for Managing Business Processes In An It Company," *Business Management, D. A. Tsenov Academy of Economics, Svishtov, Bulgaria*, issue 3 Year 2020, p. 72–90. Available at: https://ideas.repec.org/a/dat/bmngmt/y2019i3p72-90.html.
56. Zayat, W. & Senvar, O. (2020) “Framework Study for agile software development via scrum and Kanban,” *International Journal of Innovation and Technology Management,* 17(04). Available at: https://doi.org/10.1142/s0219877020300025.

Internet resources:

57. *5 lean principles every engineer should know* (2022) *ASME.* Available at: https://www.asme.org/topics-resources/content/5-lean-principles-every-should-know (Accessed: November 10, 2022).
58. 360pmo (2022) *The Evaluation to an Agile Triangle, 360PMO.* Available at: https://www.360pmo.com/evolution-agile-triangle/ (Accessed: September 22, 2022).
59. Atlassian (2023) *Jira overview: Products, projects and hosting, Atlassian*. Available at: https://www.atlassian.com/software/jira/guides/getting-started/overview#jira-software-hosting-options (Accessed: January 17, 2023).

60. Burrows, M. (2022) *Introducing Kanban through its values, Kanban Tool.* Available at: https://kanbantool.com/kanban-library/why-kanban/introducing-kanban-through-its-values (Accessed: November 10, 2022).
61. Chin, D. (2017) *How mural's marketing team uses Agile + design thinking methods to craft solutions and drive growth: Mural blog, RSS*. Available at: https://www.mural.co/blog/agile-dt (Accessed: January 10, 2023).
62. Clayton, M. (2022) *Do you understand the uole of the project manager?*, *Online PM Courses*. Available at: https://onlinepmcourses.com/understand-role-of-the-project-manager/ (Accessed: November 2, 2022).
63. Downs, J. (2020) *20 best agile project management tools, Justinmind*. Available at: https://www.justinmind.com/blog/agile-tools/ (Accessed: January 27, 2023).
64. *Excel definition: What is Excel, and what is its purpose?* (2022) *Simple Sheets - 100+ Excel & Google Sheets Spreadsheet Templates*. Available at: https://www.simplesheets.co/blog/Excel-Definition (Accessed: January 27, 2023).
65. *History* (2022) *MSD*. Available at: https://www.msd.com/company-overview/history/ (Accessed: March 7, 2023).
66. *Introduction to Microsoft whiteboard* (2022) *Microsoft Learn*. Available at: https://learn.microsoft.com/en-us/microsoft-365/whiteboard/?view=o365-worldwide (Accessed: January 27, 2023).
67. *JIRA Waterfall Model – Javatpoint* (2021) *www.javatpoint.com*. Available at: https://www.javatpoint.com/jira-waterfall-model (Accessed: December 5, 2022).
68. *Kanban Style Work Boards: Pros and Cons – Kanbanchi* (2022) *www.kanbanchi.com*. Available at: https://www.kanbanchi.com/kanban-style-work-boards (Accessed: November 10, 2022).
69. Latre, A.B. (2017) *What is a "proxy" product owner? why it is found so often?, Scrum.org*. Available at: https://www.scrum.org/resources/blog/what-proxy-product-owner-why-it-found-so-often (Accessed: December 22, 2022).
70. *Lean přístup - ManagementMania.com*. (2018) *www.managementmania.com*. Available at: https://managementmania.com/cs/lean (Accessed: October 17, 2022).
71. *Lean thinking and Practice* (2021) *Lean Enterprise Institute*. Available at: https://www.lean.org/lexicon-terms/lean-thinking-and-practice/ (Accessed: October 23, 2023).
72. Mastropasqua, F. (2021) *The Ultimate Guide to the sprint backlog, Clearly Agile*. Clearly Agile. Available at: https://www.clearlyagile.com/agile-blog/the-ultimate-guide-to-the-sprint-backlog (Accessed: February 23, 2023).
73. *Microsoft PowerPoint Slide Presentation Software: Microsoft 365* (2023) *Slide Presentation Software | Microsoft 365*. Available at: https://www.microsoft.com/en-us/microsoft-365/powerpoint (Accessed: January 27, 2023).
74. Montoya, G. (2022) *What's jira align?, DEISER Blog*. Available at: https://blog.deiser.com/en/what-is-jira-align (Accessed: January 17, 2023).
75. *MSD/Merck* (2016) *Logos Download*. Available at: https://logos-download.com/10559-msd/merck-logo-download.html (Accessed: March 7, 2023).
76. *MSD IT* (2023) *msd.cz*. Available at: https://www.msd.cz/cs/it/ (Accessed: March 7, 2023).
77. Radigan, D. (2021) *4 best practices for Sprint Planning Meetings, Work Life by Atlassian. Atlassian.* Available at: https://www.atlassian.com/blog/agile/sprint-planning-atlassian (Accessed: February 17, 2023).

78. Riaz, Q. (2022) *Agile Project Management for Microsoft Teams: Pros & Cons, PSOHUB_KeepItTogether*. PSOhub Ventures B.V. Available at: https://www.psohub.com/blog/agile-project-management-for-microsoft-teams (Accessed: January 17, 2023).
79. Rudder, A. (2022) *What is the Project Management Triangle?, Forbes*. Forbes Magazine. Available at: https://www.forbes.com/advisor/business/project-management-triangle/ (Accessed: November 7, 2022).
80. *Scaled agile solves large scale development challenges* (2019) *Agilest*. Available at: https://www.agilest.org/scaled-agile/ (Accessed: February 1, 2023).
81. Vasiliauskas, V. (2023) *Agile team roles and responsibilities: A brief guide, Teamhood*. Available at: https://teamhood.com/agile-resources/agile-team-roles/ (Accessed: January 7, 2023).
82. Waterfall methodology: Project management | Adobe Workfront (2022). Available at: https://business.adobe.com/blog/basics/waterfall (Accessed: December 5, 2022).
83. Williams, C. (2022) *5 best agile project management tools* (2022), *Accelo*. Available at: https://www.accelo.com/resources/blog/5-best-agile-project-management-tools/ (Accessed: January 10, 2023).
84. *What is a project manager? the lead role for Project Success* (2022) *CIO.* Available at: https://www.cio.com/article/230682/what-is-a-project-manager-the-lead-role-for-project-success.html (Accessed: December 17, 2022).
85. *What is Agile Software Development?* (2022) *Agile Alliance*. Available at: https://www.agilealliance.org/agile101/ (Accessed: December 12, 2022).
86. *What is Agile Project Management?* (2022) *APM*. Available at: https://www.apm.org.uk/resources/find-a-resource/agile-project-management/ (Accessed: December 12, 2022).
87. *What is mural and how to use it for design thinking (2022) Innovation Training | Design Thinking Workshops*. Available at: https://www.innovationtraining.org/what-is-mural-and-how-to-use-mural-for-design-thinking/ (Accessed: January 17, 2023).
88. *What is Kanban?* (2023) *Agile Alliance* |. Available at: https://www.agilealliance.org/glossary/kanban/#q=~(infinite~false~filters~(postType~('page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~(~'kanban))~searchTerm~'~sort~false~sortDirection~'asc~page~1) (Accessed: January 20, 2023).
89. *What is Kanban? explained in 10 minutes: Kanbanize* (2023) *Kanban Software for Agile Project* Management. Available at: https://kanbanize.com/kanban-resources/getting-started/what-is-kanban (Accessed: January 20, 2023).
90. *Your guide to agile software development life* cycles (2022) *Easy Agile*. Available at: https://www.easyagile.com/blog/agile-software-development-life-cycle/ (Accessed: December 20, 2022).

Other resources:

91. Digital.ai (2022) *15th State of Agile Report*. Available at: https://info.digital.ai/rs/981-LQX-968/images/SOA15.pdf?_ga=2.2071153.1396698264.1663000131-2013811604.1663000131 (Accessed: December 12, 2022).
92. Digital.ai (2023) *16th State of Agile Report*. Available at: https://digital.ai/resource-center/analyst-reports/state-of-agile-report/ (Accessed: January 27, 2023).
93. DORA. (2021). *The 2021 state of DevOps report*. Available at: https://cloud.google.com/devops/state-of-devops (Accessed: December 17, 2022).
94. Editor (2019) *Agile Project Management: Best practices and methodologies, AltexSoft.* AltexSoft. Available at: https://www.altexsoft.com/whitepapers/agile-project-management-best-practices-and-methodologies/ (Accessed: December 12, 2022).

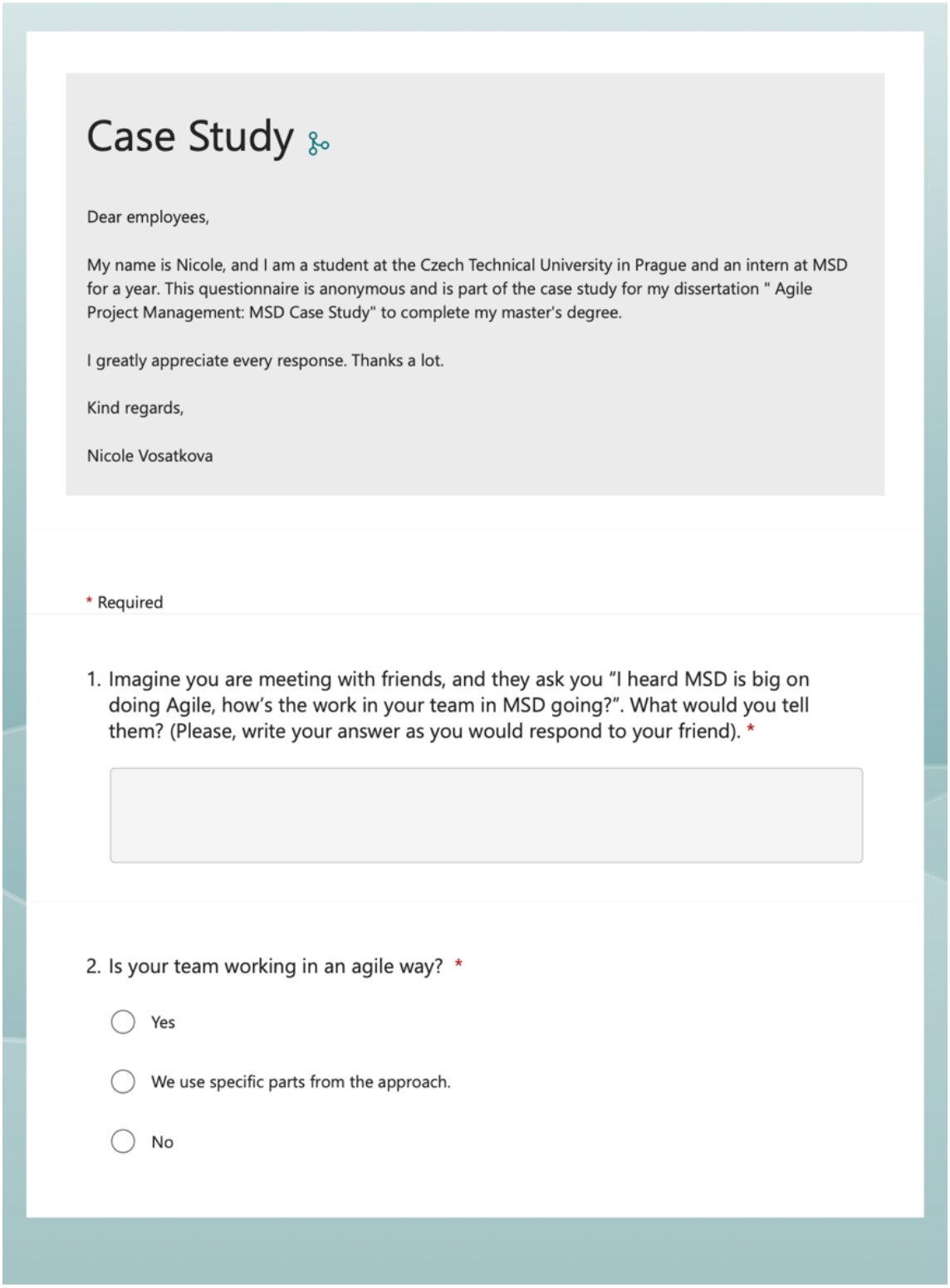
List of Figures

List of Tables

List of Graphs

Appendices

Annex 1 Questionnaire



Case Study

Dear employees,

My name is Nicole, and I am a student at the Czech Technical University in Prague and an intern at MSD for a year. This questionnaire is anonymous and is part of the case study for my dissertation " Agile Project Management: MSD Case Study" to complete my master's degree.

I greatly appreciate every response. Thanks a lot.

Kind regards,

Nicole Vosatkova

* Required

1. Imagine you are meeting with friends, and they ask you "I heard MSD is big on doing Agile, how's the work in your team in MSD going?". What would you tell them? (Please, write your answer as you would respond to your friend). *

2. Is your team working in an agile way? *

- ◯ Yes
- ◯ We use specific parts from the approach.
- ◯ No

3. What agile frameworks/methods do you use in your team? *

- [ ] Scrum
- [ ] Kanban
- [ ] Extreme Programming
- [ ] Lean Development
- [ ] Nexus
- [ ] SaFe
- [ ] LeSS
- [ ] Custom solution we have developed ourselves.
- [ ] Other

4. What agile practices do you use in your team? *

- [ ] Having Scrum Master
- [ ] Having Product Owner
- [ ] Convey information face to face
- [ ] Product backlog
- [ ] Working together daily
- [ ] Daily Stand-ups
- [ ] Working in iterations
- [ ] Sprint Planning
- [ ] Examine team's effectiveness (Review)
- [ ] Retrospectives
- [ ] Kanban board/Scrum board
- [ ] Small self-organized teams
- [ ] Feedback loops
- [ ] Outcome oriented prioritisation
- [ ] Frequent product delivery
- [ ] Customer need discovery
- [ ] Technical agility (XP practices)
- [ ] Other

5. What of the agile practices do you consider as most effective? *

Please select at most 3 options.

- [ ] Having Scrum Master
- [ ] Having Product Owner
- [ ] Convey information face to face
- [ ] Product backlog
- [ ] Working together daily
- [ ] Daily Stand-ups
- [ ] Working in iterations
- [ ] Sprint Planning
- [ ] Examine team's effectiveness (Review)
- [ ] Retrospectives
- [ ] Kanban board/Scrum board
- [ ] Small self-organized teams
- [ ] Feedback loops
- [ ] Outcome oriented prioritisation
- [ ] Frequent product delivery
- [ ] Customer need discovery
- [ ] Technical agility (XP practices)
- [ ] Other

6. What agility supporting tools do you use in your team? *

- [ ] MURAL
- [ ] JIRA
- [ ] JIRA Align
- [ ] MS Teams
- [ ] MS White Board
- [ ] MS PowerPoint
- [ ] MS Excel
- [ ] MS Planner
- [ ] Kanbanize
- [ ] Trello
- [ ] ActiveCollab
- [ ] Pivotal Tracker
- [ ] Other

7. How would you rate the benefits you see agile approach bringing to your team? *

	Poor	Below average	Average	Above average	Excellent
Increased visibility	◯	◯	◯	◯	◯
Increased flexibility	◯	◯	◯	◯	◯
Increased product quality	◯	◯	◯	◯	◯
Increased business value	◯	◯	◯	◯	◯
Increased customer satisfaction	◯	◯	◯	◯	◯
Decreased risks	◯	◯	◯	◯	◯
Better predictability	◯	◯	◯	◯	◯
Better communication within the team	◯	◯	◯	◯	◯
Better collaboration within the team	◯	◯	◯	◯	◯
Easier achievement of project objectives	◯	◯	◯	◯	◯
Faster process of work	◯	◯	◯	◯	◯
Increased speed of delivery	◯	◯	◯	◯	◯

8. What factors do you consider critical to the success of implementing an agile approach? *

	Unimportant	Of little importance	Moderately important	Important	Very important
Top management support	◯	◯	◯	◯	◯
Team capability and training	◯	◯	◯	◯	◯
Team environment	◯	◯	◯	◯	◯
Usage of agile techniques	◯	◯	◯	◯	◯
Customer involvement	◯	◯	◯	◯	◯
Project management process	◯	◯	◯	◯	◯
Organisational culture	◯	◯	◯	◯	◯
Communication	◯	◯	◯	◯	◯
Openness to change	◯	◯	◯	◯	◯

9. What impact do you see the agile development practices having on the outcomes of your work? *

10. What is the reason for not using the agile approach?

- ◯ We don't have enough knowledge about agile.
- ◯ We have tried to use agile, however, it has not been working.
- ◯ We have not even discussed the use of agile.
- ◯ No customer support of agile approach.
- ◯ I don’t know.
- ◯ Other

11. What are the most significant barriers to adopting agile practices in your team? *

Please select at most 3 options.

- [ ] Lack of wider buy-in around agile
- [ ] Resistance to change
- [ ] Lack of open communication
- [ ] Poor collaboration and knowledge sharing
- [ ] Pervasiveness of traditional development methods
- [ ] Lack of management support
- [ ] Inconsistent processes and practices across teams
- [ ] Lack of skills/experience with agile methods
- [ ] Insufficient training and education
- [ ] Organisational culture at odds with agile values
- [ ] Unwilling to admit mistakes and learn from delivery failure
- [ ] Lack of time to adapt agile
- [ ] Client's unwillingness to collaborate in agile manner
- [ ] Problems with using of agile tools
- [ ] Dependencies
- [ ] None
- [ ] Other

12. What are the expectations your customers have around the way you work? *

- [ ] They expect us to be predictable.
- [ ] They expect us to be able to change priorities quickly.
- [ ] They expect us to deliver frequently.
- [ ] They expect us to fulfil their requirements exactly.
- [ ] They expect us to work with them and regularly gather their feedback.
- [ ] They expect us to meet exact delivery dates.
- [ ] They expect us to understand their work and come up with solutions.
- [ ] I don't know.
- [ ] Other

13. What type of work is your team mostly doing? *

- ◯ Product
- ◯ Platform
- ◯ Service
- ◯ Program
- ◯ I don't know.
- ◯ Other

14. Which IT area are you a part of? *

- ◯ Animal Health IT
- ◯ CTO
- ◯ Digital MMD
- ◯ EIT
- ◯ Human Health IT
- ◯ ITRMS
- ◯ IT Strategy Realization
- ◯ MRL IT
- ◯ TIOE
- ◯ Other

Record of the thesis loan

I declare:
I give my permission for lending this thesis. The user confirms with his/her signature, that she/he will duly cite this thesis in the list of used literature.

Name and Surname: Nicole Vosátková
In Prague: April 25, 2023 Signature:

Name	Department/workplace	Date	Signature