AHMED ALKHARAM

ATILIM UNIVERSITY

MODIFYING SCRUM METHODOLOGY FOR RESEARCH-BASED SOFTWARE DEVELOPMENT PROJECTS

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
ATILIM UNIVERSITY

AHMED ALI SALEM ALKHARAM

A MASTER OF SCIENCE THESIS
IN
THE DEPARTMENT OF SOFTWARE ENGINEERING

September 2021

MODIFYING SCRUM METHODOLOGY FOR RESEARCH-BASED SOFTWARE DEVELOPMENT PROJECTS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF

ATILIM UNIVERSITY

BY

AHMED ALI SALEM ALKHARAM

IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF SOFTWARE ENGINEERING

September 2021

Approval of the Graduate School of Natural and Applied Sciences, Atilim University.

_____

Prof. Dr. Ender KESKİNKILIÇ
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of **Master of Science in Software Engineering, Atılım University**.

_____

Prof. Dr. Ali Yazıcı
Head of Department

This is to certify that we have read the thesis Modifying Scrum Methodology for research-based software development projects submitted by Ahmed Alkharam and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science- Architecture/Doctor of Philosophy.

_____	_____
Prof. Dr. Ali Yazıcı	Asst. Prof. Dr. Ziya Karakaya
Co-Supervisor	Supervisor

Examining Committee Members:

Assoc. Prof. Dr. Çiğdem Turhan Software Eng. Department, Atılım University	_____
Asst. Prof. Dr. Ziya Karakaya Computer Eng. Department, Atılım University	_____
Assoc. Prof. Dr. A. Murat Özbayoğlu Computer Eng. Department, TOBB ETU	_____

Date: *20/Sep/2021*

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: AHMED ALKHARAM

Signature:

ABSTRACT

MODIFYING SCRUM METHODOLOGY FOR RESEARCH-BASED SOFTWARE DEVELOPMENT PROJECTS

ALKHARAM, AHMED ALI SALEM
MS., Department of Software Engineering
Supervisor: Asst. Prof. Dr. Ziya Karakaya
Co-Supervisor: Prof. Dr. Ali Yazıcı

September 2021, 63 pages

Software development projects in different sizes are carried on by a group of people, who usually have a specific software development process management in order to manage their group within the project and keep track of the project process. Software development methodologies (SDM) were designed specifically for these kinds of projects. However, with different types of (SDM) there is no method precisely to manage research-based software development projects. Therefore, this study aims to conduct a systematic literature review to investigate whether Scrum methodology can be modified for research-based software development projects, identify the challenges that occurred within the projects when adopting the Scrum method along with recommended solutions for these challenges, and finally explore the best practices of Scrum method for research-based software development projects. Three main research questions were formulated to determine the information that is related to the study goals.

Keywords: Software development methodology, Research-based projects, Scrum, Software Engineering, Project Management.

ÖZ

ARAŞTIRMA TEMELLİ YAZILIM GELİŞTİRME PROJELERİ İÇİN SCRUM METODOLOJİSİNİN DEĞİŞTİRİLMESİ

ALKHARAM, AHMED ALI SALEM
MS., Yazılım Mühendisliği Bölümü
Tez Yöneticisi: Asst. Prof. Dr. Ziya Karakaya
Ortak Tez Yöneticisi: Prof. Dr. Ali Yazıcı

Eylül 2021, 63 sayfa

Farklı boyutlardaki yazılım geliştirme projeleri, genellikle belirli bir yazılım geliştirme süreci yönetimini takip eden bir grup insan tarafından, proje içindeki grubunu yönetmek ve proje sürecini takip etmek için yürütülür. Yazılım geliştirme metodolojileri (SDM) bu tür projeler için özel olarak tasarlanmıştır. Ancak, farklı türlerde araştırma tabanlı yazılım geliştirme projelerini yönetmek için geliştirilmiş özel bir yöntem yoktur. Bu nedenle, bu çalışma, araştırma tabanlı yazılım geliştirme projeleri için Scrum metodolojisinin değiştirilip değiştirilemeyeceğini araştırmak için sistematik bir literatür taraması yapmayı, Scrum yöntemi benimsenirken projelerde ortaya çıkan zorlukları ve bu zorluklar için önerilen çözümleri belirlemeyi ve son olarak Araştırmaya dayalı yazılım geliştirme projeleri için Scrum yönteminin örnek uygulamalarını belirlemektedir. Bu amaçla üç ana araştırma sorusu belirlenmiş ve sonuçlar tez kapsamında sunulmuştur.

Anahtar Kelimeler: Yazılım geliştirme metodolojisi, Araştırmaya dayalı projeler, Scrum, Yazılım Mühendisliği, Proje Yönetimi.

My dissertation is dedicated to my parents, my wife, my daughter "Leia", and my brothers and sisters, who have always been a constant source of inspiration and support, and who have taught me to work hard to pursue my dreams.

ACKNOWLEDGMENTS

I express sincere appreciation to my supervisor Asst. Prof. Dr. Ziya Karakaya for his guidance, insight, and valuable directions throughout the research. I also would like to thank my co-supervisor Prof. Dr. Ali Yazıcı for his valuable feedback and support.

I shall also express my gratitude to the respectful jury members for their valuable time and extreme patience.

Finally, I would like to thank ATILIM University for providing me with the opportunity to be a part of such a great university and for fostering such a positive learning environment. I would also like to thank the academic and administrative staff for their dedication throughout these difficult times and for helping us in this journey.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF SYMBOLS/ABBREVIATIONS

RBSDPs	Research-based software development projects
SLR	Systematic Literature Review
SDM	Software development Methodology
SDLC	Software Development Life Cycle

CHAPTER 1

INTRODUCTION

There are many different traditional software development methodologies that help to produce a software product. Each of these methodologies has its own advantages and disadvantages. None of these methodologies can cover all types of software development projects necessities. Moreover, there is no such methodology that is explicitly designed for research-based software development projects in universities, research organisations, or at any companies, despite including a variety of software development methodologies.

There is a difference between research-based software development projects and industry projects which is the purpose of the project. Research-based software development projects are mainly focused on research findings and learning new technologies or new concepts, and they usually conduct these projects as a part of their graduate research projects or funded projects by the university or any research organisations or companies, industry projects on the other hand, are concerned with delivering a working product for a profit purpose and customer satisfaction [1].

The software development methodology process is a series of steps that software engineers follow to deliver a working system. This process includes several phases, called the Software Development Life Cycle (SDLC). As mentioned above, there are many types of software development methodologies. Few among the well-known traditional plan-based methods, namely, are Waterfall, Incremental, Prototyping, and Spiral. Agile, unlike the traditional plan-based methods, is a customer-friendly and precise approach, Agile methodology has different types of methodologies, and the most well-known ones are Extreme Programming, Scrum, and Kanban.

In this research, we highlight Scrum in research-based software development projects. Scrum methodology was introduced to develop software products as it has proved to be faster and more reliable in the software engineering industry. Nevertheless, many other industries have adopted Scrum as an approach, such as construction [2],

education [3], marketing [4], military [5], and automotive [6] due to its effectiveness and efficiency in managing the project process. The research field is among them. Before Scrum was invented, most software projects were created based on the Waterfall approach. The process for this method is slow as each stage should finish first before starting the next stage towards the final step. This process often results in delays in months or years and ends up in a product with low quality that does not meet customer's requirements, and if there are changes, it will cost a fortune as they will follow the same process from the beginning.

Scrum has a simple idea, if you start a project, you should regularly check on it, and if what you are doing meets customer needs, you are in the right direction. However, Scrum has its advantages and disadvantages. Therefore, this research study is conducted by using a Systematic Literature Review (SLR) to acquire the difficulties and issues in implementing Scrum methodology on research-based software development projects, Scrum practices in different research projects, and to address the benefits of Scrum on those projects. Scrum comes with its own challenges. In past research, challenges have been analysed at the organisational and team level. However, Scrum needs to be adopted at the team's level as well. Moreover, there are other challenges such as inexperience, poor communication and specializations, lack of teamwork, risk management, documentation, and low quality.

1.1. The Significance of the Study

It is seen that no specific approach is being followed to manage projects within the research environment; researchers' teams utilize methods based on their previous knowledge on similar projects that they have been involved in. However, several methodologies have been introduced to be adopted in research-based software development projects. These methods aim to increase the efficiency of the project's process, delivering the work on time, and meeting research project requirements. Nevertheless, each of these methodologies has its flaws. Among those methods is the Agile Scrum approach. This research study will help to address the challenges faced

by research project team members when Scrum methodology is adopted in their research-based software development projects.

1.2. Research Objectives

- To explore the underlying factors that cause issues among Scrum team members.
- To identify the main issues impacting the efficiency of the research project.
- To propose recommendations to Scrum team members to overcome these challenges during projects.
- To formulate recommendations to prevent these issues from occurring before starting the project.

1.3. Research Questions

In this study, three main questions are investigated based on a systematic literature review. which are directly related to adopting Scrum in research-based software development projects. Currently, few papers have modified the Scrum method to apply to their research-based projects see [7][8][9]. This study aims to identify the issues faced by researchers when they apply Scrum method to their research-based projects and to find the best practice in Scrum to use in these types of projects. And to accomplish that, we have formulated the following research questions:

RQ1. Can research-based projects be conducted using Scrum?

As we determined in the introduction section, there is no specific method designed for research-based projects, this encourages us to ask this question in order to investigate further and find the latest results about whether Scrum can be adopted in research-based projects without any problems.

RQ2. What are the challenges involved in using Scrum Methodology in research-based software development projects?

We asked this question to identify common challenges and issues that occur when applying Scrum in a research-based software development project. And to do this, we conducted a systematic literature review as a fact-finding technique.

RQ3. What is the possibility of processes modification to conduct the Scrum approach in research-based projects?

There are a few studies that modified Scrum to be applied in research-based projects. This research question aims to investigate the literature on Scrum practices in research-based projects.

1.4. Methodology

A systematic literature review is conducted in this study based on the methodology proposed by Kitchenham [10]. The main reason for choosing a systematic literature review in this study is to summarize the existing evidence and find gaps in the current research studies related to Scrum in research projects.

1.5. Structure of this Thesis

In Chapter 2, the background of the problem is listed together with the literature review. The research methodology, the search process, selection of articles, and details about the applied methodology is explained in Chapter 3. The benefits, challenges, and practices of the Scrum methodology in the research-based software development project is stated in Chapter 4. Finally, Chapter 5 summarizes the answers to the research questions as well as the discussion and conclusion.

CHAPTER 2

LITERATURE REVIEW

2.1. Background information

Before we begin giving information about the existing literature, we started with some background information about other most well-known software development methodologies to have a clear understanding of their advantages and disadvantages. This will demonstrate the reason for choosing Agile Scrum methodology in our research study.

The software development process is a series of steps that software engineers follow to produce a product. This process includes several phases, which has another name called Software Development Life Cycle (SDLC). There are many types of software development methodologies. However, this section focuses on the main ones and talks about them briefly. These methods are Waterfall, Incremental, Prototyping, Spiral, Agile, Extreme Programming, Scrum, and Kanban.

The definition of each software development methodology, along with its strengths and weaknesses, is detailed in this section with their references.

2.1.1. Traditional Plan-based Approach

The traditional plan-based approach generally consists of sequential steps, starting with gathering the requirements, then planning, analysing, designing, implementing, testing and deployment. The traditional plan-based approach has been around for a long time, it provides a disciplined process towards software development with the purpose of making software developments products more efficient [11].

The traditional plan-based approach consists of four main characteristics: predictive approach, documentation, process oriented, and tool oriented [11].

2.1.1.1.Waterfall Model [12]

The waterfall model is a sequential approach and is sometimes called a classic life cycle. It starts with customer requirements gathering and goes through designing, implementation, verification, and finally maintenance. Each step should be finished before moving on to the next stage. It is suitable for projects with a fixed budget and well-defined requirements. See Figure 2.1.

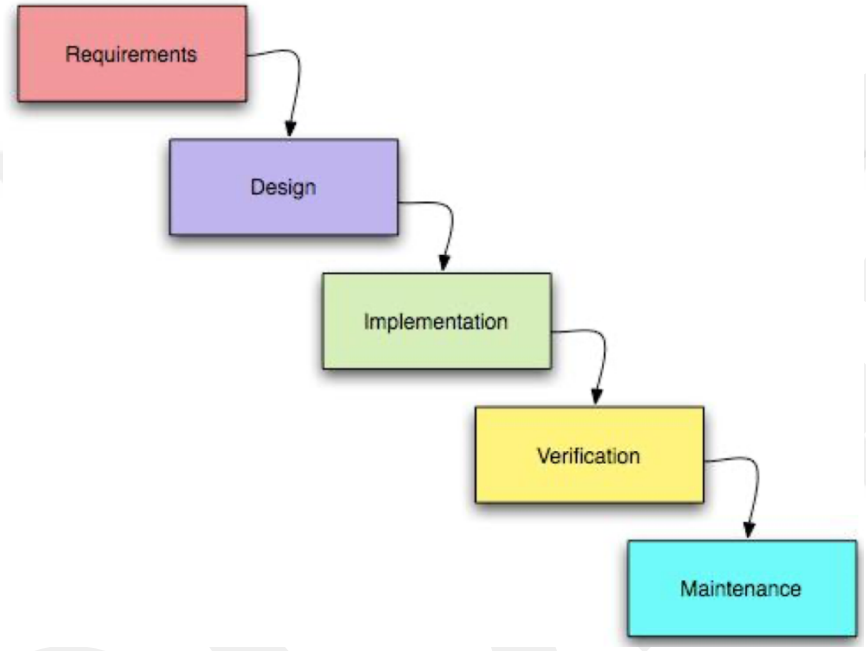

Figure 2.1 Waterfall model [12]

Advantages

- Solid documentation, as each team documents their work at the end of each stage.
- Clear requirements.
- High-quality software.

Disadvantages

- Waste of other departments resources when it is not in their stage.

- Requirements are unchangeable. If the client wants to change a requirement, the software process cycle must be finished to the last stage and then started over.
- Hard to fix an issue if a team delivers the work with defects to the next stage, which leads to more problems.

2.1.1.2. Incremental Model [13]

The incremental model combines waterfall model elements and iterative elements, it divides customers' requirements into small modules, and each module goes through the software development life cycle process from design, implementation, verification, and maintenance. Finally, it will produce deliverable increments of the software. This method is suitable for low to medium-risk projects.

Advantages

- Customers will receive a working function of their software at the end of each incremental.
- Requirements are changeable.
- Easy to test and debug each incremental separately.

Disadvantages

- It needs good planning and design.
- It needs a clear definition of the whole system before starting up.

2.1.1.3. Prototyping Model [14]

Prototyping is a technique that can be used with one of the other methods. However, it can also be a stand-alone process. The basic idea of the prototyping model is a way to identify software requirements when the customers are not sure about their system' requirements, or the customer does not know what they want exactly. In this case, the software developer identifies as many requirements as possible from the clients, they

will start to design prototypes much the same as the actual software. The prototype is deployed and evaluated by the clients, and they can provide their feedback.

Advantages

- Customers will be involved in the product development process.
- Errors occur in the early stages.
- It helps to find out which features are missing.

Disadvantages

- Designing a prototype consumes more time.
- Customers might think the working prototype is the actual delivered product.

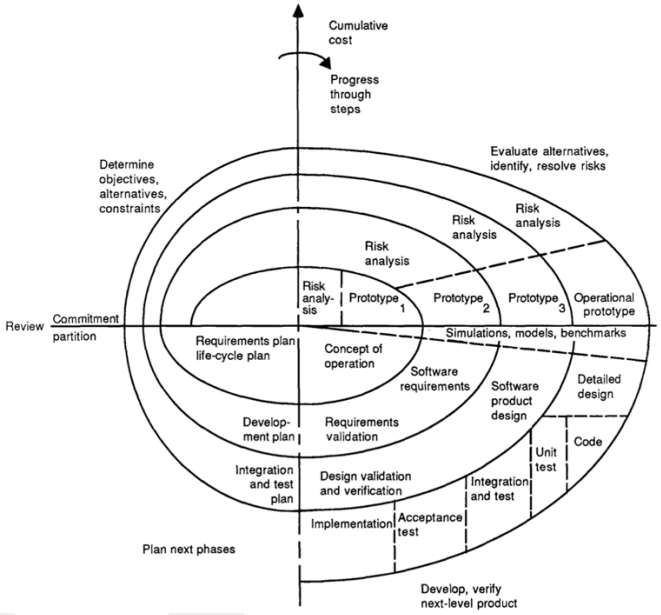

Figure 2.2 Spiral model [15]

2.1.1.4. Spiral Model [15]

The spiral model is an iterative process. Barry Boehm [16], who proposed the spiral model, has described it as a risk-driven process. It is suitable for projects with risks, especially for military projects.

The process goes in a clockwise direction and starts from the centre, where the requirements plan stage continues until the implementation stage. In each stage, the risk will be evaluated. See Figure 2.2.

Advantages

- Risks are well analysed.
- Strong documentation.
- New functions can be added.

Disadvantages

- The cost of this process is high.
- Time-consuming in risk evaluation.

2.1.2. Agile Approach [17]

Traditional plan-based Software development methodologies focus on upfront planning, gathering requirements, and initiating the budget. Agile approach gives more importance to teamwork, customer collaboration, and delivering a working system over documentation, as stated in Agile manifesto [18]. Agile has several methodologies. In the following section, we only listed the three most famous Agile methods, Extreme Programming, Scrum, and Kanban.

2.1.2.1 Extreme Programming “XP” [19]

Extreme programming is one of the Agile methodologies. It consists of four main activities, namely planning and managing, designing, coding, and testing. In planning and managing, the customers and developers work together to group the requirements into user stories and prioritize them based on customers' requests. In the coding stage, developers start to develop several unit tests then start to code as pair programmers.

Two developers work together; one writes the code, and the other one reviews the code.

Extreme programming is suitable only for software development projects and the main reason of creating XP was to response to requirements changes and produce high-quality products [11]. It was added in this section just to demonstrate different Agile methodologies.

Advantages

- Customers and developers work together, which leads to customer satisfaction.
- Errors in coding are avoidable, as one developer is focuses on detecting errors in his co-worker codes.
- Requirements are changeable.

Disadvantages

- The cost of this process is high.
- More work for developers.
- Lack of documentation.

2.1.2.2 Scrum Method [20]

Scrum is one of the Agile methodologies created by Jeff Shtherland and Ken Schwaber in 1993. "Scrum is an iterative, incremental framework for projects and product or application development." [20].

In Scrum, the software is delivered in increments called "Sprints" usually, and it takes from 1-4 weeks for each sprint [20]. Each sprint begins with planning and ends with a review. The sprint - planned by the Scrum team - is a time-boxed meeting, which could last up to 4 hours to prioritize items that will be selected during the meeting. The items should be completed by the end of the sprint. A daily Scrum meeting by a Scrum team is 15-minutes long, and each team member addresses three questions: what did I do yesterday? What will I do today? And what impediments are in my way? The reason for this is to inspect the sprint progress, and when the sprint is completed, the team reviews the sprint with the stakeholders, and if it meets their needs, the Scrum team

and product owner emphasize that this sprint is "Done". Scrum encourages stakeholders to be part of the project, so the stakeholder who is called "Product owner" is attending sprint review meetings to review "Sprint backlog". The retrospective meeting is scheduled to help the teamwork in the completed sprints. Scrum produces three artefacts, named: product backlogs, sprint backlogs, and burn-down charts. Backlogs contain customer requirements, and daily burn-down charts show the cumulative work remaining. Key roles, artefacts, and events are summarized in Figure 2.3.

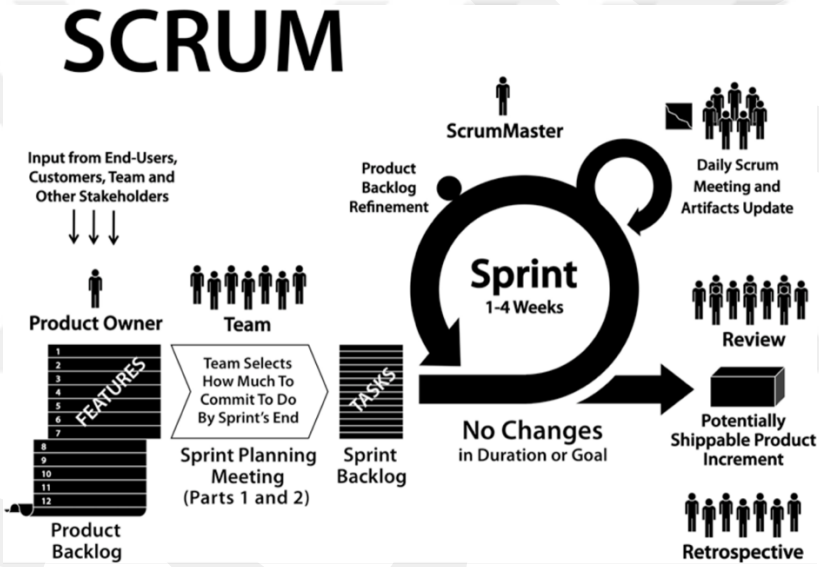

Figure 2.3 Scrum model [20]

Advantages

- Customers and developers work together.
- Customer satisfaction.
- It is flexible.
- Projects are easily managed.
- Requirements are changeable.

Disadvantages

- Scrum teams require training.
- Lack of documentation.

2.1.2.3 Kanban Method [21]

Kanban was developed by Toyota to control the inventory in the factory with minimum use of resources. Kanban means “single card” in Japanese; Kanban methodology uses sticky notes on whiteboards to control the work process, it helps to visualize the workflow [21].

David J. Anderson, Kanban founding father, has created Kanban method based on Incremental and Evolutionary process. The fundamentals of Kanban method are based on two types of principles: Change management principles and Services delivery principles. Change management principles means start with what you do now, agree to pursue incremental and evolutionary change, and encourage acts of leaderships at all levels. Services delivery principles focus on customer needs, managing the work and regularly reviewing the network services [21].

Advantages

- Increased Visibility of the Flow.
- Improved Delivery Speed.
- Improved Predictability.
- Increased Customer Satisfaction.

Disadvantages

- Board can be complicated.
- Lack of timing, there is no timeframe for each phase.

Kanban is good for manging the workflow, minimizing the time cylce and organizing it. But, research-based software development projects contain a number of team members and each has a different role and responsiablity, Scrum however, is created to help manage the work process and structure team members.

2.1.3 Research-based Software Development Project Characteristics

Frascati Manual by OECD [22] has provided a good definition of Research and development projects; “Research and development (R&D) comprise creative and systematic work undertaken in order to increase the stock knowledge including knowledge of humankind, culture and society. And to advise new application of available knowledge.” [22] Research and Development projects are aimed at new findings.

In order for the activities to be (R&D), they need to meet five important criteria set by OECD in Frascati Manual, which are: that the activity must be novel, creative, uncertain, systematic, and transferable and/or reproducible. **Novel** is aimed to new findings. **Creative** is based on originality, but with new a concept to improve an existing knowledge. **Uncertain** which means to be uncertain about the outcome. **Systematic** means it should be conducted in a planned way. **Transferable and/or Reproducible** means the result of the result can be possibly reproduced. (R&D) activities can be Basic research, Applied research and Experimental Development. **Basic Research** is theoretical work to gain new knowledge of a phenomena. **Applied Research** is original examination to gain new knowledge, it is primarily aimed towards a practical goal. **Experimental Development** “is systematic work, drawing on knowledge gained from practical experience and producing addition knowledge” [22]. Those are the characteristics of research-based software development projects which are based Frascati Manual [22].

2.2. Adopting Scrum for Research-based Software Development project Methodology

We have divided this section into three subsections, each subsection is related to one of the research questions.

2.2.1. Adopting Scrum in research-based projects

Scrum methodology has been a valuable approach to many projects. It has helped overcome many issues delivering high-quality products. What lead Professors Michael Hicks and Jeffrey S. Foster to adopt Scrum methodology in their research project [9] is that their traditional approach reached its limit. As the more researchers joined their research team, the quality of collaboration among the team decreased [9]. By modifying the Scrum approach to their objectives and calling it (SCORE) then adopting it in their research project, they managed to increase collaboration among team members [9].

A study held by Marchesi, M., Mannaro, et al [7] has reported that they adopted Scrum method to manage their research project due to its ability to manage complex projects and changing requirements.

Another study by Ota, M. [8] reported using Scrum methodology to guide collaboration among academic projects because of its strict agreement on timing and budgeting.

Some researchers have stated in their study Lima, I. R., de Castro Freire, et al [23] that the reason they have chosen Scrum method is that before adopting Scrum, they faced issues such as project following-up, deadlines, and lack of communication between teams and clients. In order to overcome these problems they started to conduct Scrum in their research project and had a positive impact.

According to Hidalgo, E. S. [24], 85% of data science projects have failed to meet customer's expectations due to lack of project methodologies. In addition, they stated that other studies show the success factor of data science projects using Scrum methodology. Thus, the authors have decided to develop Scrum method specifically for data science projects named (Scrum-DS).

Another group of researchers have developed a method based on Scrum methodology to manage a research project's process called "ARDev". They chose Scrum because of the way it manages the project, adding value to their research team, and enhancing team collaboration, as reported in their study Bezerra, D. R., Dias-Neto, A. C., & da Silva Barreto, R [25].

[26] have proposed a new methodology that suits research-based projects, called RBAgile which is based on Agile Scrum method, and its principles are based on suggestions of interviewers who have worked in research-based projects using a different approach which encounters a number of issues related to problems with people, budget problems, and communication problems.

2.2.2. Challenges in adopting Scrum in a research-based project

Scrum methodology has been an essential factor in the success of many projects in the software engineering field and other science fields, as mentioned in Chapter 1. However, Scrum comes with its challenges. In the following review, nine papers have reported challenges related to adopting Scrum.

Hicks, M., & Foster, J. S. [9] stated that their junior researchers were complaining about not following up with the research topic, plus not gaining experience from senior researchers due to the time limitation during statue meeting "15 minutes daily meeting".

Collaboration among team members is one of the Agile manifesto "Customer collaboration over contract negotiation" [18] and an essential aspect in Scrum methodology. However, Lima, I. R., de Castro Freire, et al [23] reported that one of the issues they faced when they adopted Scrum was lack of collaboration, as some team members were not sharing ideas with the others and refused to cooperate. Another study Hanslo, R., & Mnkandla, E [27] stated that the reasons that lead Scrum team members to face lack of collaboration are that they do not receive any good feedback, and products are not matching the requirements.

Hanslo, R., & Mnkandla, E [27] have categorized several challenges under "Global Scrum adoption challenges" based on 22 publications from 2008 to 2017. The most frequent one of these challenges is the lack of project management and topic knowledge/skills. Paulk, M. C. [28] stated that it is important to invest in training your team members as Scrum requires highly qualified members. The study of Akif, R., & Majeed, H. [29] stated that 50% of Scrum team members have no proper training.

One of the agile values is "Working software over comprehensive documentation" [18]. Scrum shares the same values. Still, it remains a significant issue for many Scrum adopters according to Akif, R., & Majeed, H. [29]. Based on the survey they conducted in their study, it is hard to track changing requirements as they come from different sources such as emails which is hard to follow.

As mentioned above, collaboration is an essential factor in Scrum methodology, which helps team members to share ideas and progress on their tasks. Nonetheless, this can arise an issue regarding cultural differentiation when there are researchers who have different backgrounds or cultures. In the study Paulk, M. C. [28] stated that culture challenge is causing an issue in adopting Scrum at the team level.

Another important factor in Scrum method is communication between team members, therefore, one of the reasons software projects failed are due to lack of communication López-Martínez, J., Juárez-Ramírez, R., Huertas, C., Jiménez, S., & Guerra-García, C. [30]. Another study reported that communication is a challenge on the team level Hanslo, R., Mnkandla, E., & Vahed, A. [27]. They stated communication is an essential component between team members to finish a task "Sprint".

2.2.3. The process of adopting Scrum in a research-based project

Scrum methodology was designed to manage software development projects, and it has been involved in other industries projects too. Scrum is a flexible approach and can be modified to be suitable to any other field.

In Scrum, there are three core roles, namely, product owner, Scrum Master, and Scrum team. Additionally, there are three Scrum artefacts, product backlog, sprint backlog, and increment. In their research, Marchesi, M., Mannaro, K., Uras, S., & Locci, M. [7], they have modified Scrum roles and Scrum artefacts to fit their project's objectives, in that event, two other roles have been added; Unit Coordinator and Research Unit. Unit Coordinator, is the one who makes sure that the output of the artefacts is meeting the required deliverables whereas the research unit is a group of researchers who have a responsible role for features and deliverables. Also, two artefacts have been added; Impediment List, which means a list of anything that will hinder the project's quality and Burndown Graphic, which is basically an Agile tool; the goal of this tool is to show the project's remaining time of work in sprint in graphic.

As for Hicks, M., & Foster, J. S. [9] they were more interested in Scrum events such as Sprint meetings, Sprint reviews, and daily meetings. Thus, they initiated two main events; Status meetings which are similar to the daily meeting but are only conducted three days a week and On-demand Technical meetings, which are between students and their advisor discussing their research works, issues, and obstacles.

Another study conducted by Bezerra, D. R., Dias-Neto, A. C., & da Silva Barreto, R. [25], who developed a method based on Scrum called "ARDev", which is devoted to research projects, it has an additional role called Research Starter compared to Scrum core roles. The process of the method is both iterative and incremental, which has three phases; Conception, Construction, and Evaluation.

CHAPTER 3

RESEARCH METHODOLOGY

The study is concerned with the clarification of the research area that is being studied for identifying whether Scrum can be applied in research-based software development projects, its benefits, challenges, solutions, and practices of Scrum in research projects. The direction of this research in which the study was carried out is to identify the problems related to Scrum in projects that happened in the research environment. With the identification of the research problem, the research is directed towards the solutions that can be provided by this study.

The research method that is applied in this research study is a systematic literature review SLR, which means conducting a secondary study based on other primary studies (Scholarly articles, research papers, and published studies) to find any gaps and to purpose more investigation and provide solutions. The main purpose behind the adoption of SLR is that current information about adopting Scrum in research-based software development projects, its challenges, and practices need to be analysed. SLR tries to check, examine, and explain a specific phenomenon based on research questions from other current research studies.

We have adopted systematic literature review as a method in our study because the current data about adopting Scrum methodology on research-based software development projects need to be further investigated. In addition, SLR is a clear methodology with straightforward steps where the results will possibly be unbiased.

The secondary source of data collection in this study is what was directly collected by the author himself and mainly collected through semi-structured interviews. In this study, four interviews were held, with individuals who have worked on different research-based software development projects using Scrum methodology, the interview questions are in Appendix B.

An invitation was sent to different Scrum communities on Facebook, LinkedIn, and Reedit. As well as Scrum users in person via email. Unfortunately, only four responded and participated in the study. The participants are from different countries, two of them are Scrum Masters and the other two are Scrum Team Developers. All the participants have experience in using Scrum in three projects or more.

In this chapter, we will be discussing the kind of methodology that we have adopted. And to conduct this research properly, we have followed Kitchenham and Charters guidelines [31] which contain several activities such as identifying research questions, developing a review protocol, defining search strategies, study selections, quality assessment, and data extraction.

3.1.Research Design

There are different types of research designs , and to classify them based on the research purpose and method, those research designs are categorized under qualitative and quantitative designs. Qualitative design is mostly used to collect non-numerical data such as open-ended questions. The quantitative design is used in close-ended questions, and the results can easily be transformed into numerical data. This research study uses qualitative design since it involved collection of secondary research studies in addition to open-ended questions.

3.2. Research Questions

The most critical part of the systematic literature review is defining the research questions. This study focuses on adopting Scrum in research projects, highlighting the challenges that come with it and the process of adopting it. And to be able to achieve this, we formulated the following research questions:

RQ1. Can research-based projects be conducted using Scrum?

RQ2. What are the challenges involved in using Scrum Methodology in research-based software development projects?

RQ3. What is the possibility of processes modification to conduct the Scrum approach in research-based projects?

The first question is mainly to check whether Scrum can be applied in a research-based project or not. The second question is to identify issues that occur when Scrum is being adopted in research-based software development projects. The last question is to address the process of adopting Scrum in research-based projects.

3.3. Search Strategy

After we formulated the research questions, the next step was to search for research papers related to our research questions. The procedures that have been followed to identify the related studies are:

i. We have looked for software engineering digital libraries.
ii. We have constructed search strings using keywords and Boolean ANDs and ORs.
iii. We have evaluated the found paper's abstract, keywords, and title based on inclusion and exclusion criteria.
iv. We have documented all papers that met all the set criteria in our created database.
v. We have read the chosen papers in-depth to re-evaluate them.
vi. We have kept the relevant papers and excluded the irrelevant ones.

We started to search for primary studies from five digital libraries such as:

- Google Scholar https://scholar.google.com
- IEEE Explore https://ieeexplore.ieee.org/
- SpringerLink https://link.springer.com
- Research Gate https://www.researchgate.net
- ScienceDirect https://www.sciencedirect.com

The following is the search string, and the keywords that have been used in searching for papers are:

- (Scrum) AND (research) AND (project)
- (adopting) AND (Scrum)
- Scrum in research
- Adopting Scrum framework
- Scrum in a research project.

3.4. Specifying inclusion and exclusion criteria

Inclusion and exclusion criteria aim to get the most related papers to the research questions.

Table 3.1 Inclusion and Exclusion Criteria

Inclusion Criteria	Exclusion Criteria
1. The papers should be in English	1. Any other studies that are not in English
2. The publication date of the papers should be between 2005 and 2021	2. Studies that are about scrum in general
3. The studies must be about scrum in research projects.	3. Studies that are not related to research questions
4. The studies can be both primary and secondary.	

3.5. Study Selection

In this section, we will be explaining the process of choosing studies that will be answering our research questions. Selected papers have gone through the inclusion and exclusion process that is mentioned in the previous section 3.3.

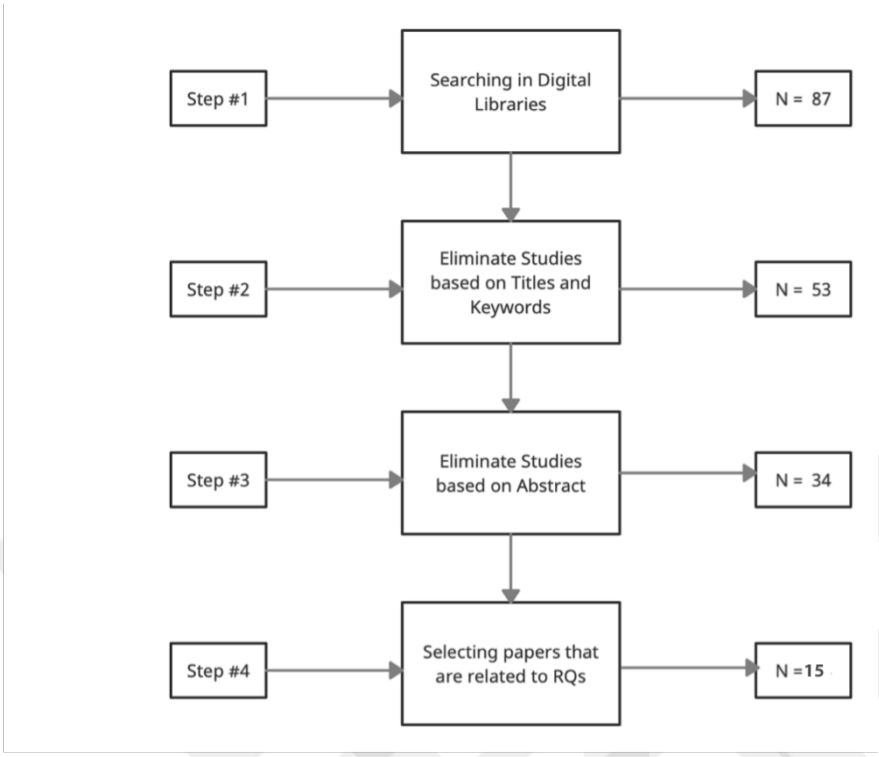

Figure 3.1 The Selection Process

We first started to search in digital libraries using search strings and keywords; as was explained in section 3.2, We found 87 papers in the most well-known scientific research libraries, next; we evaluated the title and keywords based on the inclusion and exclusion criteria, which reduced the papers to 53, again we reviewed the abstract, reducing to 34 papers, finally, after reading all 34 papers in detail, we obtained 15 papers that are relevant to our research question. Figure 3.1 shows the selection process. And Table 3.2 shows the selection of the papers based on each library. Appendix 1 contains all of the selected papers in detail.

Table 3.2 Selected Papers

Library	First Search	Title & Keywords	Abstract	Full review
Google Scholar	31	11	10	7
IEEE	21	15	12	4
SpringerLink	13	9	3	2

Research Gate	15	13	7	1
ScienceDirect	7	5	2	0
Total	87	54	34	15

3.6. Quality Assessment

The quality assessment section aims to validate the quality of the studies that have been found and check whether the selected papers are biased or not according to the inclusion and exclusion criteria section. Therefore, it leads us to ask questions that are recommended by Frank's [32], such as:

- Are the study's goals clearly stated?
- Have the research questions been answered?
- Are the data collections well carried out?
- Are the statistical and research methods described?
- Have the negative findings been recorded?

3.7. Data Extraction

Once the eligibility criteria were applied and the paper selection process was completed, each paper was read in more detail and data was extracted. An excel spreadsheet was used as an extraction form, and data was recorded manually in there. From each paper, we recorded the following information:

• Article’s title

• Aim of study

• Authors

• Authors’ affiliation

• Year of publication

• Journal/Conference name

• Name of the practice Methodology

• Name of challenges

• Findings

3.8.Ethical considerations

This research study does not contain any ethical issues since it is based on investigating the existing studies. Thus, the examination of secondary studies tends to reduce bias within the research study.

CHAPTER 4

BENEFITS, CHALLENGES, SOLUTIONS, AND PRACTICES OF SCRUM METHODOLOGY IN RESEARCH-BASED PROJECTS

In this chapter, we will be explaining the challenges, solutions, benefits, and best practices of adopting the scrum framework on research-based projects. This information can help us to answer our research questions. The first section will address the benefits of using scrum in research-based projects. The results of this section will help us answer the first question of our research questions. The second section will explain the challenges of using Scrum in research-based projects and its solutions to overcome those issues, and this section will help us answer the second research question. The last section will summarize the best practices of adopting Scrum in research projects, and this section offers the answer to our last research question.

4.1 Benefits

This section will help us answer the first of our research questions; the research question is:

RQ1. Can research-based projects be conducted using Scrum?

4.1.1 The benefit of adopting Scrum in research-based projects.

There are many factors involved in researchers and organizations adopting scrum in their research projects over other methodologies. We summarized all of them in this section, and before we go through them in detail, we have listed them all in Table 4.1 with their references.

The following table contains the main benefits of using Scrum in research-based projects.

Table 4.1 Benefits of using Scrum in research-based projects.

Benefit	Reference
1. Group contact creates a meaningful community. 2. Team members' communication improvement. 3. Monitoring research progress is transparent. 4. It responds to changes in the project quickly. 5. High collaboration between team members to produce successful research results. 6. Daily Project Follow-up. 7. Collaboration with the clients. 8. Frequent delivery to the clients.	• Hicks & Foster [9] • Marchesi, Michele, et al, [7] • Ota & Martin, [8] • Lima IR, de Castro Freire T, Costa HA. [23] • Hidalgo & Senabre, [41] • Bezerra DR, Dias-Neto AC, da Silva Barreto R. [25] • Santos, Nuno, et al. [39] • Baijens J, Helms R, Iren D, [34]

Group contact creates meaningful community

What makes Scrum one of the best approaches is its meeting activities; Scrum allows team members to connect during the meetings, gain experience from their success and failures, and helps create a community with a sense of momentum and accomplishment. Furthermore, it builds confidence among junior researchers by spending more time with senior researchers for gaining knowledge and experience Hicks & Foster, [9].

Team members communication improvement

Another key factor of Scrum's meetings is communication among team members. Meetings in Scrum have increased the communication within the group, and colleagues have become more interested in other team member's task progress. In addition, constant communication can help the team come up with the best decision

solutions. Hicks & Foster, [9]; Marchesi, Michele, et al, [7]; Baijens J, Helms R, Iren D, [34].

Monitoring research progress is transparent

Transparency is one of the important aspects of Scrum methodology; Scrum imposes transparency inside and outside the team, which helps everyone to understand what is going on in each sprint. Plus, it increases trust among team members. Ota & Martin, [8].

It responds to changes in the project quickly

Almost all projects tend to have changes in requirements while they are in the implementation process. Yet, most of the traditional software development processes with fixed budgets face a hard time dealing with this issue which results in costing them more money and time. However, the Scrum approach is based on feedback and adjustments, and it is in its nature to respond to changes in project requirements, technologies, and organizations. Responding to changes, means customers satisfaction, increasing the chances of the project's success. Marchesi, Michele, et al, [7]; Lima IR, de Castro Freire T, Costa HA, [23]; Hidalgo & Senabre, [24].

High collaboration between team members to produce successful research results

Another important key factor of Scrum methodology is the collaboration between team members. High collaboration helps the team to increase the level of management efficiency and to support successful research productions and frequent deliveries of research outcomes. Hidalgo & Senabre, [24]; Bezerra DR, Dias-Neto AC, da Silva Barreto R. [25].

Daily Project Follow-up

The important factor of Scrum methodology is a daily follow-up of the project. One of the Scrum activities is the daily Scrum meeting. Each day the Scrum Master and team members conduct a stand-up meeting for daily project updates and setting plans for the day's tasks. The Scrum meeting usually is conducted in the morning and is time-boxed to 15 minutes, where each team member is required to answer three

questions; What have they done yesterday, what will they be doing today, and what kind of issues have they faced. Marchesi, Michele, et al, [7]; Lima IR, de Castro Freire T, Costa HA, [23]; Baijens J, Helms R, Iren D, [34].

Collaboration with the clients (Stakeholder)

Alongside the importance of collaboration among team members, the collaboration between the team and clients is equally important. In the Scrum approach, the client is involved throughout the entire project, works closely with the Scrum master, and stays engaged in all Scrum meetings. The client prioritizes product requirements in the product backlog; in every sprint review, the client can demonstrate a working functionality of the product. As a result of this, customer satisfaction and high-quality product are enhanced. Santos, Nuno, et al. [39].

Frequent delivers to the clients

Scrum is an iterative and incremental based approach. We will summarize the Scrum process to have an understanding image of the whole process. It starts where the product owner (stakeholder) creates the product backlog that contains lists of requirements, the chosen items from the backlog will get into an event called the sprint that has a time-box approximately from two weeks to a month, at the end of the sprint a product increment is delivered to the client for approval. If it is approved, a new product backlog is set to be prepared for the next iteration. Therefore, customers will be able to receive working features and give their feedback on the product quickly. This also gives the client the opportunity to have an idea of what the final product will look like. Baijens J, Helms R, Iren D, [34]; Marchesi, Michele, et al, [7].

4.2 Challenges

This section will address the challenges in the Scrum research project. The challenges in this section will be divided into three sections; the first section describes the challenges at the organization level, the second section will be about challenges at a team level, and the last section will describe challenges at the people level. The information in this section, will help us answer the second research question, which is:

RQ2. What are the challenges involved in using Scrum Methodology in research-based projects?

4.2.1 Challenges at the organization level

The successful adoption of Scrum depends on several factors, these factors are at the organization, team, and people level. In this subsection we will be addressing organisational challenges that have been extracted from the selected papers in our research study. Before we got into details of those challenges, we listed them in Table 4.2 with their references.

Table 4.2 Challenges at the organization level.

Challenges	Reference
1. Organizational/Institutional culture does not align with Scrum work. 2. Lack of management support.	• López-Martínez, Janeth, et al. [30] • Hanslo, R., Mnkandla, E., & Vahed, A. [40] • Hanslo, R., & Mnkandla, E. [27] • Hidalgo, Enric Senabre. [24]

Organizational/Institutional culture does not align with Scrum work

Organizational/Institutional culture is one of the key factors that cause resistance in adopting Scrum methodology. The team can be Scrum but will face a major issue if the organization/institution is not; the motivation to conduct a research project will be hard to manage. Hidalgo, Enric Senabre. [24]; López-Martínez, Janeth, et al. [30].

Lack of Management support

Management support helps organizations/institutions to adopt new methodologies in a positive perspective, and this will create an innovative environment. If an organization lacks management support, it will create an innovation hostile environment. In addition, management support has a direct impact on the adoption of new ideas. Hanslo, R., & Mnkandla, E. [27].

4.2.2 Challenges at the team level

This subsection will highlight the challenges that occurred at the team level in adopting Scrum methodology in research-based projects. Table 4.3 contains all the extraction data with their references.

Table 4.3 Challenges at the team level.

Challenges	Reference
1. Limitation of team number. 2. Team member collaboration. 3. Communication issues among team members. 4. Lack of trust. 5. Documentation issues. 6. Information sharing issue.	• Hicks & Foster, [9] • Lima IR, de Castro Freire T, Costa HA, [23] • Hidalgo & Senabre, [24] • López-Martínez, Janeth, et al. [30] • Hanslo, R., Mnkandla, E., & Vahed, A. [40] • Hanslo, R., & Mnkandla, E. [27] • Akif, R. and Majeed, H., [29]. • Luz, M., Gazineu, D., & Teófilo, M. [41]

Limitation of team number

Scrum has a limited number of members in a team; as stated in Scrum guidelines, Scrum teams contain ten members or fewer, including Scrum Master and product owner [33]. Still, this is a challenge to many Scrum adopters with large team members that are involved in a project. Another issue with the limited number of team members is that daily meetings cannot handle a large group. Hicks & Foster, [9].

Team members collaboration

Collaboration between team members is an important factor in Scrum methodology, as it enhances the exchange of experience and knowledge among team members. Yet,

this seems to be a major challenge in adopting Scrum methodology in the research-based project. It is due to the fact that some team members prefer to work individually rather than working with a team and also to lack of commitment and transparency. Hanslo, R., & Mnkandla, E. [27]; Lima IR, de Castro Freire T, Costa HA, [23]; Hidalgo, Enric Senabre. [24]; López-Martínez, Janeth, et al. [30].

Communication issues among team members

Communication among team members and clients is a critical factor in Scrum to finish any project task. Lack of effective communication can result in project failure as it is a way of exchanging information and knowledge between team members and clients. In addition, it has a negative effect on the adoption Scrum. Hanslo, R., & Mnkandla, E. [27]; López-Martínez, Janeth, et al. [30].

Lack of Trust

As mentioned in the previous two subsections, communication and collaboration among team members and clients are important factors in Scrum methodology. However, what makes those two critical factors among the challenges is the lack trust. Trust is initiating slowly between team members as well as their relationship. Hidalgo, Enric Senabre. [24].

Documentation issues

Documentation is where all project requirements, design, testing, and other data are. In traditional methodology, documentation is critical, and it must be done in each process phase. However, Scrum has the idea that working software is more important than documentation. Still, documentation has booked itself in the Scrum adoption challenge; if a Scrum team is not using an appropriate tool that tracks the changes in requirement, they will face documentation issue. Akif, R. and Majeed, H., [29]; Hanslo, R., & Mnkandla, E. [27].

Information sharing issue

Information sharing is an important factor during collaboration among team members, and this helps the team to work more efficiently and gain more experience through

knowledge-sharing with their colleagues. But this seems to be an issue with individuals who have a problem working in groups due to lack of trust between team members. Lima IR, de Castro Freire T, Costa HA, [23]; Luz, M., Gazineu, D., & Teófilo, M. [41].

4.2.3 Challenges at the people level

Adopting new methodologies has an impact on the behaviour of people. This subsection will address the challenges that occurred at the people level when adopting Scrum methodology in the research-based projects. Table 4.4 contains a number of challenges that were extracted from the selected research papers with their references.

Table 4.4 Challenges at the people level.

Challenges	Reference
1. Lack of experience. 2. Lack of training.	• López-Martínez, Janeth, et al. [30]. • Hanslo, R., Mnkandla, E., & Vahed, A. [40] • Hanslo, R., & Mnkandla, E. [27] • Hidalgo, Enric Senabre. [24] • Lima IR, de Castro Freire T, Costa HA, [23] • Akif, R. and Majeed, H., [29] • Paulk, Mark C. [33]

Lack of Experiences

An experience means a person has a piece of knowledge and special skills on a specific subject or area. Thus, in order to adopt new methodologies, it requires experience to work with this method. Lack of experience means lack of knowledge of a method's process. Scrum users have faced this challenge which sometimes results in misunderstanding on how to deal with customer "Product owner" requirements Hanslo, R., & Mnkandla, E. [27]; Lima IR, de Castro Freire T, Costa HA, [23].

Lack of training

Adopting Scrum means working on complex, innovative products and delivering them in a timely manner with high quality. Most Scrum adopters have no previous experience in the Scrum process, and all the gained knowledge and experience they have gathered is either from one of their team colleagues or Scrum masters. Lack of training can be a negative effect of method adoption. Therefore, proper training is required to the team members before adopting the Scrum method, to help them learn about the working of Scrum process, team rights, and enabling new techniques Akif, R. and Majeed, H., [29]; Hanslo, R., & Mnkandla, E. [27]; Paulk, Mark C. [33]; Hanslo, R., Mnkandla, E., & Vahed, A. [40].

4.2.4 All Challenges/Issues with their descriptions

This section shows the challenges/issues which occurred when adopting Scrum for research-based software development projects or other kinds of software development projects, Table 4.5 contains all the challenges/issues with their description and whether they occurred in RBSDPs or any other software projects.

Table 4.5 Challenges with descriptions.

Challenges/Issues	Level	Description	Occurred at	
			RBSDP	**All kind of software projects**
Organizational/Institutional culture does not align with Scrum work.	Organization	Team project can be Scrum but will face issues if the organization culture is resistant to changes.	✓	✓
Lack of management support	Organization	Adopting new technology or method needs management support otherwise it is doomed to fail.	✓	✓
Limitation in the number of team members	Team	Scrum has a limited number of members in a team, and when exceeding the limited number, issues may occur.	✓	
Team members collaboration	Team	Lack of collaboration between team members may lead to significant errors in applying Scrum methodology	✓	✓
Communication issues among team members	Team	Lack of effective communication can result to project failure.	✓	✓

Lack of trust	Team	Lack of communication and collaboration sometimes is the result of a lack of trust between team members.	✓	✓
Documentation issue	Team	Documentation is one of the common Scrum challenges, as Scrum has the idea of working software is more important than documentation.	✓	✓
Information sharing issue	Team	Information sharing is an important factor during collaboration, but still happens due to some team members preferring to work individually.	✓	✓
Lack of experience	People	Lack of experience means a lack of knowledge of a method's process.	✓	✓
Lack of Training	People	Most Scrum adopters have no previous experience in how Scrum works, thus they need to be trained to work with Scrum.	✓	✓

4.2.5 Recommended Solutions for Scrum Challenges in research-based project

Limitation of team number: the Scrum team has a limited number of members as listed in the Scrum guide "The Scrum Team is small enough to remain nimble and large enough to complete significant work within a Sprint, typically 10 or fewer people." [33]. But some projects have large sized teams which is an issue in Scrum as it is hard to conduct the daily meetings with large sized teams as well as controlling them at once, the solution to this issue was stated in [35] which they called "Scrum of Scrums" which basically means if the Scrum team was larger than ten members, the team has to be divided into two teams with the respect to limit of seven to nine in each team, this process continues, if the number of team members is higher than twenty, they have to be divided into three teams and so on the higher the number of team members gets.

Collaboration and Communication issues: collaboration and communication challenges among team members and stakeholders are based on a number of factors, culture/people difference, team size issue, lack of commitment in meetings, and sometimes distance differences if the teams work in different places, so in order to increase collaboration and communication, team members should gather regularly, have obligatory presentations for all team members, social/friendly activities to get to know each other, and as for distance differences using synchronous and asynchronous tools can be helpful. !!

Lack of trust: building and maintaining trust between team members is an important factor to the success of the project, and what causes trust issues is poor team bonding, lack of cultural understanding, ineffective communication, and missing face-to-face interaction [36], and in order to overcome lack of trust issue, it is important to begin activities and lunch meetings in order for the team members to get to know each other and to build an initial sense of trust between team members [37].

Documentation issue: documentation in software development takes huge efforts and time from the project timeframe in order to do it properly as each stage should be

documented before it goes to the next one, however, one of Agile manifesto is stated "Working software over comprehensive documentation" [18] which means to reduce the amount of documentation within the project and to produce working software. Documentation should not be an issue, another Agile principle could help us overcome this challenge "At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly." [18], the team should often inspect whether this project needs to be documented or not, if yes how much documentation is needed, and of course they could document what they need in order to use it for their intended purpose, and all of this could be made during the Sprint Retrospective.

Information sharing issue: information sharing issue occurred at the team level due to lack of trust among team members which leads to some of them not sharing information with others. Even though Scrum method encourages information sharing during the whole cycle and work progress [38], information sharing between team members improves the productivity with quality, therefore, to solve this issue, Scrum master has to motivate their team members by providing rewards, some people expect something in return for their knowledge sharing. This is important as increasing trust between team members will encourage them to work together more effectively.

Lack of experience and Lack of Training: lack of training and lack of experience are a common issue when newly adopting Scrum methodology in a project, and it is sometimes due to either the project's low budget, releasing planned product over training, or expecting the employee may leave the company. Instead of focusing on the reasons listed above companies/organizations should lean towards training their employees as they are a valuable asset to their establishment. Investing in training programs increases their loyalty and skills.

4.3 The practice of Scrum in research-based projects

In this section, we will be highlighting different Scrum practices that have been applied to research-based projects, this will help us understand the distinction between each practice. Those practices will help us answer our last research question:

RQ3. What is the possibility of processes modification to conduct the Scrum approach in research-based projects?

The practices that are demonstrated in this section have been extracted from the selected research papers in our study. The Table 4.6 contains each practice name with its reference.

Table 4.6 Practices of Scrum in research-based projects.

Practice	Reference
SCORE (based on Scrum method)	Hicks, M., & Foster, J. S. [9].
EURACE Scrum	Marchesi, Michele, et al. [7]
Scrum-DS	Baijens, J., Helms, R., & Iren, D [34]
ARDev	Bezerra, D. R., Dias-Neto, A. C., & da Silva Barreto, R. [25]
RBAgile	İbrahim Cereci, & Ziya Karakaya. [26]

Before we go into details of these practices, let's have a quick look at Scrum method practice. It begins with Scrum team, which includes Scrum Master, Product Owner, and Scrum Developers team.

Scrum Master

This is the person who serves the product owner, scrum developers' team, and the origination; he/she is the one who is coaching the team on self-management, ensuring that they are following Scrum guidelines, and helping to overcome the obstacles within the Scrum team's progress. In addition, he/she is the one working with the product owner to manage products backlog [33].

Product owner (stakeholder): Is the one who is responsible for product requirements, creating, ordering, and product backlog items. The product owner is involved throughout the entire project [33].

Scrum Developers team: They are the ones who are responsible for the sprint plan, sprint backlog and ensuring product quality. Each scrum team member requires specific skills depending on the project domain type [33].

Second Scrum has a number of events such as the sprint, Sprint planning, Daily Scrum, Sprint Review, and Sprint Retrospective.

The sprint: The sprint has a time-box from two weeks to a month; during this time, the Scrum developers' team is working on product backlog items that have been prioritized by the product owner [33].

Sprint Planning: Sprint planning is where all Scrum teams are planning and discussing the most important product backlog items, and which will be included in the next sprint.

Daily Scrum: The idea of daily Scrum meetings is to check the progress of the sprint goal and to reduce the complexity and overcome any issue during the sprint. The daily Scrum is a 15-minute event [33].

Sprint review: The purpose of Sprint review is to check the result of the sprint; Scrum developer team and product owner meet to check what has been done during the sprint [33].

Sprint Retrospective Sprint Retrospective is creating a plan to improve the quality and the effectiveness of the next sprint [33].

Next, we will be addressing the Scrum practice in the research-based project.

4.3.1 SCORE (based on Scrum method) [9]

Professor Michael Hicks and Professor Jeffrey S. Foster have adopted Scrum to manage their research group and to help solve issues that have been occurred with their old approach. They adopted Scrum with some modifications and called it SCORE, which stands for (Scrum for Research).

The practice of SCORE is similar to Scrum with modifications on Scrum events, as their main reason to adopt Scrum is because of Scrum meetings events. The modification events are:

Status Meetings: Status meetings are almost identical to Scrum daily meetings, only daily meetings occur on a daily basis; status meetings, on the other hand, occur three times a week; Tuesdays, Wednesdays, and Fridays. It carries the same activities as Scrum daily meetings.

On-demand Technical Meetings: is a meeting between researchers and their advisor to discuss technical issues, research questions and anything related to the research topics in details. This meeting is only requested by students for their lack of experience or due to an obstacle that has been reported during the status meetings. On-demand meetings are similar to sprint meetings.

Table 4.7 Advantages and disadvantages of SCORE.

Advantages	Disadvantages
• Status meetings help their team to be up to date on their project progress. • Meetings are more productive than they used to be. • It has increased communication and collaboration among team members.	• Meeting's time is not enough for their over number team members (14 members). • Status meetings do not help the junior researcher to gain experience from senior researchers.

4.3.2 EURACE Scrum [7]

EURACE is a research project aimed at developing an agent-based software platform for European economic policy design with heterogeneous interaction agents. The project was complex, and changes in project requirements may have occurred during the project process. In order to avoid any difficulty, the team adopted the Scrum

method to manage their research project with some modifications. We will be only mentioning the modified factors from the Scrum method, since the rest of it is similar to Scrum process. The following are the modified Scrum roles, events, and artefacts in the EURACE project:

Unit Coordinator: A new role added by the EURACE research team to help deliver research tasks to the team members and to coordinate the Scrum backlog items with the project owner

Unit Members: Unit members are similar to the Scrum developer team, but each has their own research unit that they belong to.

Research Unit: A number of people working together on the same site; each has a specific task in the project along with the Unit Coordinator.

Table 4.8 Advantages and disadvantages of EURACE Scrum.

Advantages	Disadvantages
• Increased communication and collaboration among team members. • Project objectives are clear. • A culture of success has been initiated among team members.	• No issue has been recorded.

4.3.3 Scrum-DS [29]

A data science group have designed a whole method from Scrum to manage data science research projects and called it (Scrum-DS). To overcome the issues that happened with their old method, this method uses Scrum elements and has been evaluated and demonstrated by experts to see if it fits their data science research project's objectives. After evaluating Scrum artifacts, roles, and events; all seems to be applicable to their data science project except for the sprint event, which has been

divided into two; the first is sprint zero for data planning, and the second one is the regular sprint event which is a time-boxed event to deliver an increment.

This practice has not recorded any advantages or disadvantages; it has only evaluated the Scrum methodology and formulated it to fit their data science research projects.

4.3.4 ARDev [25]

ARDev is a methodology that is based on Scrum philosophies to manage research projects and to support all activities are related to research management of software technology. ARDev stands for (Agile Research Development) and they have chosen Scrum due to its project management supportive and on team self-organizing. The following will be the modified practice of ARDev methodology's roles, events, and artefacts:

Research Starter (RS): is the one who is responsible for research stories and arranging them, and ensuring the team delivers the research items. **ARDev Master (AM):** Is responsible for managing the research project process, assistant to the research team, and helps them to overcome any issues. Basically, it has the exact same role as Scrum master, but it has a different concept. **Research Team (RT):** Research team is similar to the Scrum developers' group, which are those who are responsible for delivering research outcomes at the end of each sprint iteration. **Stakeholders (S):** Are the product owners who are project sponsors, or the ones who will be benefited from the project.

ARDev methodology has its iterative and incremental events; **Conception:** Is where research project and planning happen, and it is called the pre-production phase. **Construction:** This is similar to the sprint phase, it is where the research team delivers incremental research items, and it lasts between two to four weeks and is called the production phase. **Evaluation:** is where the ARDev teams evaluate the research result of the last iteration and provide recommendations for the next one. Figure 4.1 demonstrates the ARDev process.

Table 4.9 Advantages and disadvantages of ARDev.

Advantages	Disadvantages
• Frequent research items deliveries. • Increased communication and collaboration among team members. • Project objectives are clear.	• No issue has been recorded.

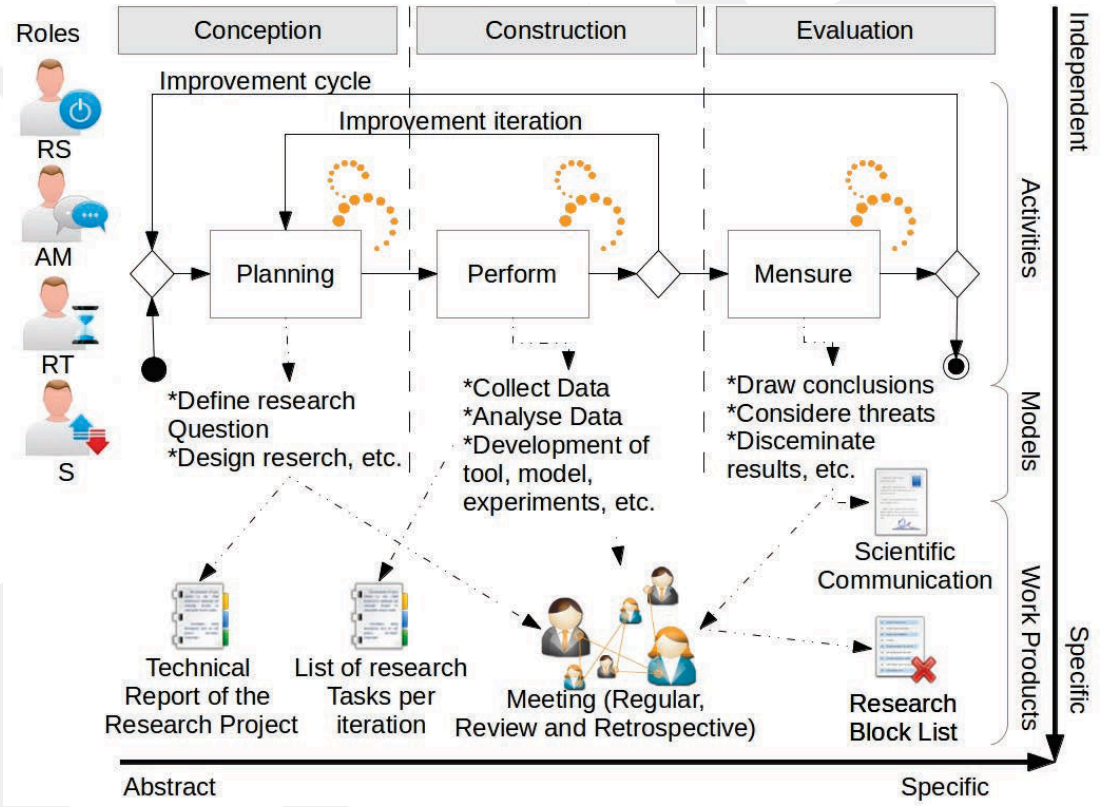

Figure 4.1 The process of ARDev method [17]

4.3.5 RBAgile [26]

RBAgile is based on Agile methodology, and it is called “Research-Based projects Agile Software Developments”, It was created because the authors, who have been involved in a number of research-based projects, interviewed a group of researchers who have similar experience in these projects. The interviewees addressed the issues

they faced while using other methodologies and suggested different solutions. Therefore, the authors have introduced this method.

The practice of this method is suitable for all projects that take place in universities, and it has many different roles, **Project Manager:** Is the leader of the project, it is an academician. He/she is the one who is responsible for project initiations, planning, designing, and deployment. **Researchers:** Are academicians who have finished their PhDs, and they are responsible for research issues, keeping track of students' progress, teaching and helping the students. **Graduate Students**: Are Master or Ph.D. graduate students and are the ones who develop the research-based project. **End-user:** Are the ones who will benefit from the outcome of the project and are responsible for project requirements and providing feedback on the product. **Customer (Product owner):** Is similar to end-user but is involved in the entire project.

There are other roles such as Area Experts (Subject Matter Experts, Domain Experts), Funding Agencies (Project Sponsor), University, Industry Partners, and Partner Institutions. The life cycle of this method is in Figure 4.2 in detail.

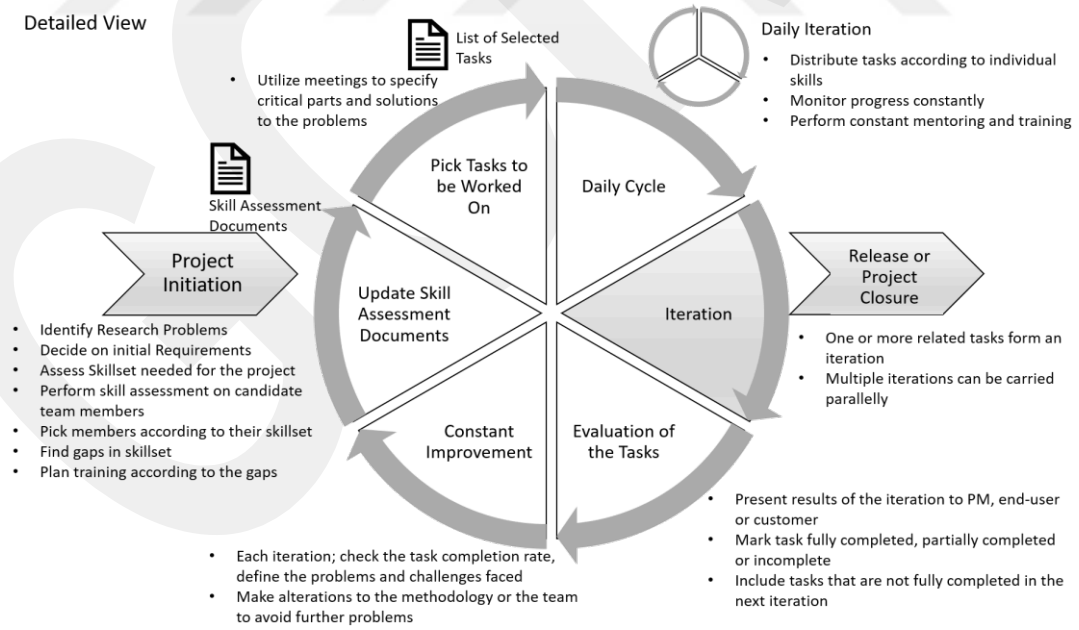

Figure 4.2 The life cycle of RBAgile [26]

4.4 Findings Discussion

In order to achieve this study's goal and fulfil its research objectives, three research questions have been formulated in Chapter 1 and have been answered in detail in Chapter 4. The research method that has been used to extract the information from the selected studies and help us to answer our research questions was a systematic literature review.

In this section, we present a list and summary of our findings, moreover, the analysis of the answers to our research questions.

The first research question is:

RQ1. Can research-based projects be conducted using Scrum?

To answer the first research question, a systematic literature review was conducted. It is seen that most of the research-based software development projects have no common software development methodology. The answer to this research question can be found in Chapter 2 and Chapter 4. The literature review in Chapter 2 involves former studies [7][8][9][23][24][25] and [29], which shows how Scrum methodology has been applied in many research-based projects. In Chapter 4 a list of benefits is stated on how the Scrum method has impact research-based projects positively. Based on this investigation, "Can research-based projects be conducted using Scrum?" can be answered as "Scrum can be applied in research-based projects".

The second research question asks:

RQ2. What are the challenges involved in using the Scrum Methodology in research-based projects?

The challenges that are involved in adopting Scrum methodology in research-based projects were gathered from the literature. In Chapter 2, literature studies [9] [23][24] [33][29][27] and [30] refer to the issues that occurred during the adoption of Scrum in research projects. In Chapter 4, the challenges have been categorized into three groups,

namely issues at the organization level, problems at the team level, and issues at the people level. Two critical challenges are recorded at the organization level, which is a lack of organizational support, and organizational culture is not aligned with the Scrum method. Six significant issues have been found at the team level, namely limitation of team numbers, team members collaboration, communication issues among team members, lack of trust, documentation issues, and information sharing issues. Based on these findings, this research question is considered answered in this study.

Challenges at the organization level

In this section, we analysed our findings on issues that occurred at the organization level, and it shows that both challenges are critical issues that have appeared in 5 studies. The following figure shows the number of studies that mentioned challenges at the organization level.

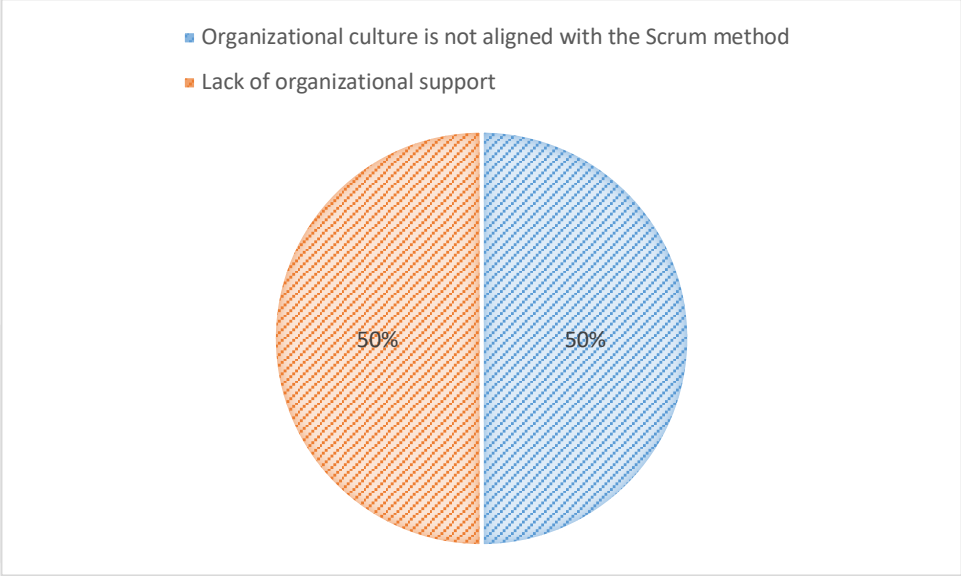

Figure 4.3. Frequency of studies on challenges at the organizational level

Challenges at the team level

In this section, we analysed our finding on issues that occurred at the team level, and it shows that the most critical issue that has appeared in nine studies for the team level is collaboration and communication issue. The following figure shows the number of studies that mentioned challenges at the team level.

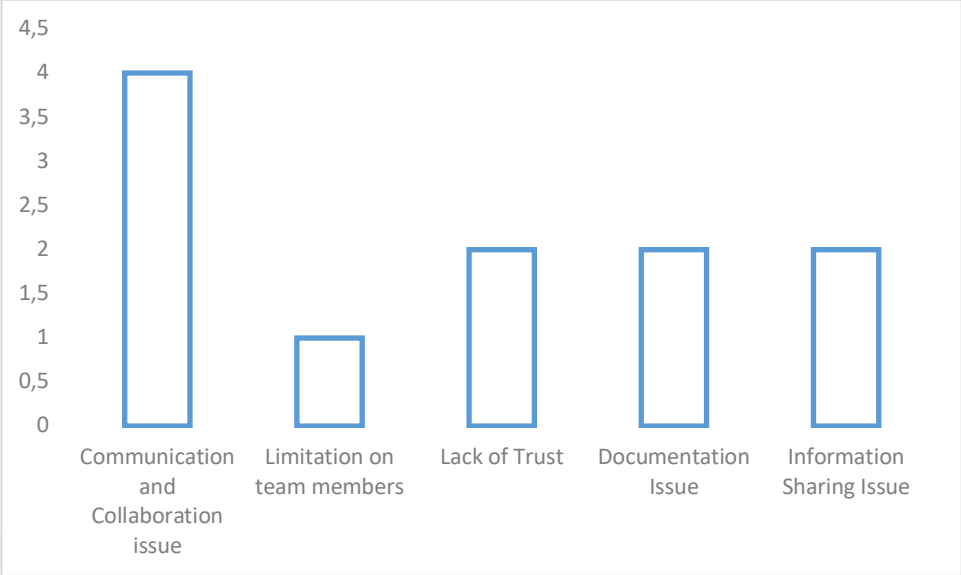

Figure 4.4. Frequency of studies on challenges at the team level

Challenges at the people level

In this section, we analysed our findings on issues that occurred at the people level, and it shows that the most critical issue that has appeared in 7 studies is lack of training. The following figure shows the number of studies that mentioned challenges at the people level.

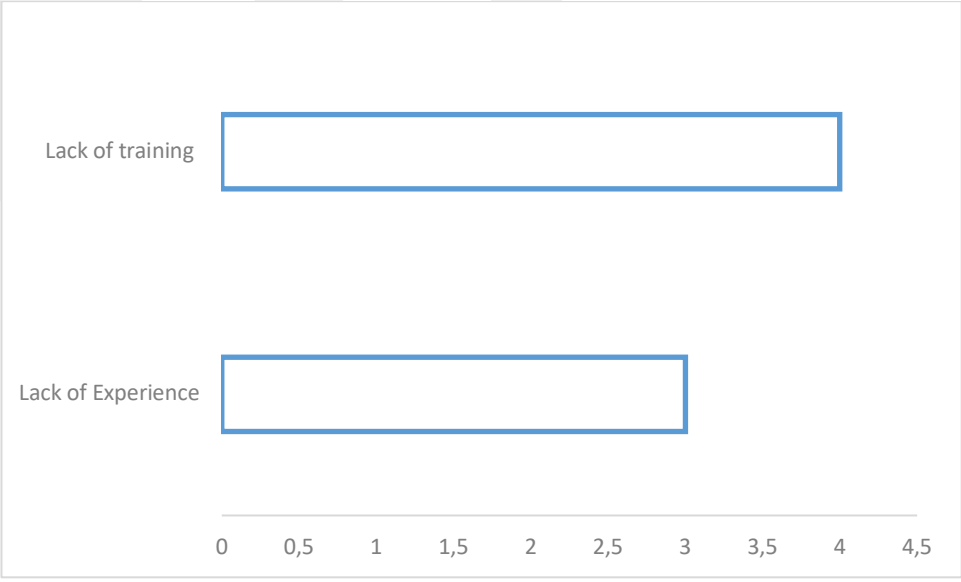

Figure 4.5. Frequency of studies on challenges at people level

The last research question was:

RQ3. What is the possibility of processes modification to conduct the Scrum approach in research-based projects?

The answer of this question can be found in Chapter 2 and Chapter 4. In Chapter 2, the literature review, the modification processes of conducting scrum methodology in research-based projects are listed in [7][9][25][26] and [29]. In Chapter 4, different Scrum methodology practices in research projects are listed with their advantages and disadvantages. Also, in Chapter 4 positive factors that come with the adoption of Scrum are stated. Accordingly, the last research question is fully answered in this research study.

4.5 Analyzing interview answers

4.5.1 Open-ended Questions

The reason behind using open-ended questions is to encourage the participants to explain or answer their questions freely based on their feelings and observations on the research issue. In this interview, we have seven main open-ended questions that require more details on the matter. The method that was followed to analyze open-ended questions is content analysis.

4.5.1.1 Content Analysis

“Content analysis is a method for finding valid and replicable evidence from textual data to their context, describing a family of analytical approaches ranging from intuitive (interpretative) analysis to systematic (strict textual) analysis.” [42]. Therefore, we organized the questions with participants’ answers in different excel sheets. Then we categorized the answers to add some formulas in each category based

on participants' answers. Figure 4.6 shows a content analysis of open-ended question one.

Open-ended question #1			
Have you and your team members had experiences with Scrum?	Yes	No	Total Response Category Count
Total respondents who answered question #1	4		4
% of respondents who answered question #1	100%	0	
Answers during the interview			
Interviewee #1: Yes	1		
Interviewee #2: Yes	1		
Interviewee #3: Yes	1		
Interviewee #4: Yes	1		

Figure 4.6 Content Analysis of open-ended question one

The first question was on whether the interviewees had experience with Scrum before they got involved into research-based software development projects. It was asked because one of the main challenges that was found in this study was that users lack experience with Scrum. However, based on the interviewee's answers, the data shows that all of them had Scrum experience or knowledge as shown in Figure 4.6.

Open-ended question #2			
Did your team follow Scrum characteristics in all stages of the research project?	Yes	No	Total Response Category Count
Total respondents who answered question #2	2	2	4
% of respondents who answered question #2	50%	50%	
Answers During the interview			
Interviewee #1: No, Some primary of Scrum characteristics, 50%		1	
Interviewee #2: No, it is hard to follow Scrum all characteristics in research environment		1	
Interviewee #3: Yes, 100%.	1		
Interviewee #4: At the critical times in the project and overall yes, my team is utilizing Scrum Agile methodologies	1		

Figure 4.7 Content Analysis of open-ended question two

The second question was on whether the team followed all Scrum characteristics or not, the interviewees data shows that 50% have followed all Scrum characteristics, as the other 50% have only modified Scrum based on their project's needs as shown in Figure 4.7.

Open-ended question #3				
1. Was there any type of collaboration challenges between team members, or between team members and stockholders?	Yes	No	Total Response Category Count	Reasons of the issue
Total respondents who answered question #3	4		4	
% of respondents who answered question #3	100%			
Answers During the interview				
Interviewee #1: Yes, some. stakeholders are not committed to the meetings. Even if they were online and it's hard to get the requirements, it's hard for team members to cmmit to the daily meetings.	1			Lack of experience, Lack of commitment
Interviewee #2: Yes, it's hard to meet the stakeholders regularly.	1			Lack of commitment
Interviewee #3: Yes	1			Lack of commitment
Interviewee #4: Yes, it exists between clients and our team.	1			Lack of commitment

Figure 4.8 Content Analysis of open-ended question three

The third question was directly related to one of the challenges that occurs when adopting Scrum which is collaboration between Scrum team and the stakeholders, the data shows that all the interviewees have faced this challenge when adopting Scrum, and normally because the stakeholder has no experience with Scrum or is not committed to it as shown in Figure 4.8.

Open-ended question #4				
1. Was there any type of communication challenges between team members, or between team members and the stockholders?	Yes	No	Total Response Category Count	Reasons of the issue
Total respondents who answered question #4	3		4	
% of respondents who answered question #4	75%	0		
Answers During the interview				
Interviewee #1: Pass				
Interviewee #2: Yes, Meeting, languages	1			Lack of meeting, Language barrier
Interviewee #3 Yes, Engineers prefer to work alone	1			Lack of trust
Interviewee #4 Yes, It happens between team members when we have multiple projects at one time.	1			Work overload

Figure 4.9 Content Analysis of open-ended question four

The fourth question was also directed to one of the challenges in adopting Scrum, and it was about communication issues between team members and stakeholders, the data shows that 75% of the interviewees' answers agreed that this is a challenge they face, whereas one participant reserved from answering the question. The main reasons of communication challenges between team members and stakeholders are language barrier, lack of meetings, lack of trust, and work overload, see Figure 4.9.

Open-ended question #5				
1. Did you face lack of trust among your team members?	Yes	No	Total Response Category Count	Reasons of the issue
Total respondents who answered question #5	1	3	4	
% of respondents who answered question #5	25%	75%		
Answers During the interview				
Interviewee #1: No		1		
Interviewee #2: No		1		
Interviewee #3: Yes	1			New engineer
Interviewee #4: No		1		

Figure 4.10 Content Analysis of open-ended question five

The fifth question was about lack of trust between team members which occurs sometimes when using Scrum as it's appeared in [24] and based on the interviewees answers only 25% agreed that this issue appeared due to having new engineers who don't have a good relationship with the other team members, whereas 75% of the interviewees answered that they have not experienced such challenge, see Figure 4.10.

Open-ended question #6			
1. Was the limitation in Scrum team members an issue?	Yes	No	Total Response Category Count
Total respondents who answered question #6	3	1	4
% of respondents who answered question #6	75%	25%	
Answers During the interview			
Interviewee #1: No		1	
Interviewee #2: Yes	1		
Interviewee #3: Yes	1		
Interviewee #4: Yes	1		

Figure 4.11 Content Analysis of open-ended question six

The last question was about whether the limitation of Scrum team members was an issue or not. We found that 75% of the interviewees agreed that it was, in fact, a challenge they faced, while 25% disagreed, see Figure 4.11.

The interviews that were conducted supported the data collected using SLR, the findings in both methods were very similar especially regarding the two most common challenges that are faced when using Scrum which are communication and collaboration issues. In addition, another challenge was also an important issue which is the limitation of team members, we found from using SLR and from the interviews that this challenge faces many Scrum users. However, some issues that were found to be common using SLR were not a problem that faced the interviewees, such as, lack of experience and lack of trust. Likewise, there were challenges mentioned during the interviews that were not found when conducting SLR, such as, lack of commitment to Scrum characteristics and language barriers.

CHAPTER 5

CONCLUSION AND FUTURE WORK

The main goal of this study is to investigate the existing studies that have adopted the Scrum methodology in their research-based software development projects and to explore the different practices that each study chose to adopt the Scrum method. Furthermore, we aim to identify the challenges that have an impact on research-based software development projects at the organization, team, and people level. In addition to the benefits that comes with Scrum method on the project processes and, lastly, to identify different Scrum practices and propose solutions to overcome the challenges.

5.1 Conclusion

As a result of this research study as well as the analysis from the systematic literature review and the interviews, it can be stated that Scrum methodology can be applied in research-based software development projects. Scrum brought many benefits to the research project team and to the project process. Besides these benefits, there are many challenges that come with Scrum and research team members must deal with them. The most important of which are communication and collaboration among team members. Fortunately, Scrum can still help overcome these issues with the effectiveness of the Scrum meetings events.

Furthermore, Scrum helps teams to deal with other challenges, such as changes in requirements and delays in delivering products, as stated in this research study. Nevertheless, the Scrum method brings additional benefits such as customer satisfaction, better product quality, and project process transparency. In addition to these benefits, adopting Scrum methodology in the research-based software development project can help the research team to gain many benefits such as enhancing collaboration and communication, higher team morale, and self-organized team members. Scrum is a flexible approach and can be modified to work with any research project domain.

5.2 Research Limitation

The plan when conducting this research study was interviewing a large number of Scrum users who have experience in research-based software development projects. The lack of such organizations and the lack of time led us to interview only four Scrum users and depend more on the studies that were available in the literature.

5.3 Future Work

The following can be potential research that can extend this study:

- Designing a survey for software engineering researchers on their current software development methodology and listing its challenges and benefits.
- Introducing the Scrum method to research-based projects and addressing its challenges, benefits, and best practice.
- Investigation and evaluation of the case studies about adopting Scrum methodology on research organizations.

REFERENCES

[1] Dias, M., Kodikara, N., & Ekanayaka, Y. "Differences between universities and industry in software development." Presented at *32nd Natl. Inf. Technol. Conf.* Colombo, Sri Lanka, pp. 207–211, 2014.

[2] Streule, T., Miserini, N., Bartlomé, O., Klippel, M., & De Soto, B. G. "Implementation of scrum in the construction industry". *Procedia engineering*, Vol. 164, pp. 269-276. 2016.

[3] Baham, C. "Implementing scrum wholesale in the classroom," *Journal of Information Systems Education*, vol. 30(3) pp, 141. 2019.

[4] Izvercianu, M., & Buciuman, C. F."An agile approach for measuring the performance of a marketing system," In *7th European Conference on Innovation and Entrepreneurship* pp. 324-331. September 2012.

[5] de Sá, F. R., de Resende Lucas, E. L., & de Oliveira, A. D. "Scrum in a strongly hierarchical organization," *In Brazilian Workshop on Agile Methods,* pp. 97-102. Springer, Cham. October 2018.

[6] Böhmer, A. I., Hugger, P., & Lindemann, U. "Scrum within hardware development insights of the application of scrum for the development of a passive exoskeleton," In *2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, pp. 790-798. IEEE. June 2017.

[7] Marchesi, M., Mannaro, K., Uras, S., & Locci, M. "Distributed Scrum in research project management," In *International Conference on Extreme Programming and Agile Processes in Software Engineering,* pp. 240-244. Springer, Berlin, Heidelberg. June 2007.

[8] Ota, M. "Scrum in research," In *Cooperative Design, Visualization, and Engineering*, pp. 109-116, Springer, Berlin, Heidelberg. 2010.

[9] Hicks, M., & Foster, J. S. (September 2010). “Adapting Scrum to Managing a Research Group,” [Online]. Available: https://drum.lib.umd.edu/handle/1903/10743 [March 23, 2021]

[10] Keele, S. “Guidelines for performing systematic literature reviews in software engineering,” *EBSE Technical report*. Vol. 5. 2007.

[11] Chaudhari, Chouksey, & Lonkar. “Traditional approach to Agile approach in Software Development,” *International Journal of Advance Research in Science and Engineering*, Vol. 07, pp. 212-218. February 2018.

[12] Mahalakshmi, M., & Sundararajan, M. “Traditional SDLC vs scrum methodology–a comparative study,” *International Journal of Emerging Technology and Advanced Engineering*, Vol. 3(6), pp. 192-196. 2013.

[13] Alshamrani, A., & Bahattab, A. “A comparison between three SDLC models waterfall model, spiral model, and Incremental/Iterative model,” *International Journal of Computer Science Issues (IJCSI),* Vol. 12(1), pp. 106. 2015.

[14] Kunicina, N., Zabasta, A., Patlins, A., Bilic, I., & Peksa, J. “Prototyping process in education and science,” *In 2020 IEEE 61th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON),* pp. 1-6. IEEE. November 2020.

[15] Krishnan, M. S. “Software development risk aspects and success frequency on spiral and agile model,” *International Journal of Innovative research in computer and communication Engineering*, Vol. 3(01), pp. 301-310. 2015.

[16] Boehm, B. W. “A spiral model of software development and enhancement,” *Special Interest Group on Software Engineering,* vol. 11(4), pp. 14-24. August 1986.

[17] Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. “Agile software development methods: Review and analysis,” *VTT electronics*, Vol.478. p. 107. 2002.

[18] “Manifesto for Agile Software Development,” Internet: https://agilemanifesto.org , [April 7, 2021].

[19] Fojtik, R. “Extreme Programming in development of specific software,” *Procedia Computer Science*, Vol. 3, pp. 1464-1468. 2011.

[20] Sutherland, J., & Schwaber, K. “The Scrum Papers. Nuts, Bolts and Origins of an Agile Process, “[Online] Available: https://www.academia.edu/download/64196137/scrumpapers.pdf [April 7, 2021] 2007.

[21] “Kanban Explained for Beginners | The Complete Guide,”. https://kanbanize.com/kanban-resources/getting-started/what-is-kanban#the_kanban_method [August. 15. 2021]

[22] OECD (2015), Frascati Manual “Guidelines for Collecting and Reporting Data on Research and Experimental Development,” *The Measurement of Scientific, Technological and Innovation Activities, OECD Publishing,* Paris. 2015

[23] Lima, I. R., de Castro Freire, T., & Costa, H. A. X. “Adapting and using scrum in a software research and development laboratory,” *Revista de Sistemas de Informação da FSMA*, Vol. 9, pp. 16-23. 2012

[24] Hidalgo, E. S. “Adapting the scrum framework for agile project management in science: case study of a distributed research initiative,” *Heliyon,* Vol. 5(3), e01447. 2019

[25] Bezerra, D. R., Dias-Neto, A. C., & da Silva Barreto, R. “ARDev: a methodology based on scrum principles to support research management on software technologies,” *In Proceedings of 24th Annual International Conference on Computer Science and Software Engineering*, pp. 363-366. November 2014.

[26] İBrahim Cereci, & Ziya Karakaya. “A Novel Software Development Methodology for Research-Based Software Projects” PhD thesis, Atilim University, Turkey. February 2019.

[27] Hanslo, R., & Mnkandla, E. "Scrum adoption challenges detection model: SACDM," *In 2018 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 949-957. IEEE. September 2018.

[28] Paulk, M. C. "On empirical research into scrum adoption," *Institute for Software Research, Carnegie Mellon University*. 2011.

[29] Akif, R., & Majeed, H. "Issues and challenges in Scrum implementation," *International Journal of Scientific & Engineering Research*, Vol.3(8), pp.1-4. 2012.

[30] López-Martínez, J., Juárez-Ramírez, R., Huertas, C., Jiménez, S., & Guerra-García, C. "Problems in the adoption of agile-scrum methodologies: A systematic literature review," *In 2016 4th international conference in software engineering research and innovation (conisoft)*, pp. 141-148. IEEE. April 2016.

[31] Kitchenham, B., & Charters, S. "Guidelines for performing systematic literature reviews in software engineering," *EBSE Technical report*. Vol. 5, Ver. 2.3. 2007.

[32] Fink, A. *Conducting Research Literature Reviews: From the Internet to Paper*. Los Angeles, USA, 2019, pp 140-235.

[33] Ken Schwaber, Jeff Sutherland. "The Scrum Guide". https://scrumguides.org/scrum-guide.html , 2020, [June 23, 2021].

[34] Baijens, J., Helms, R., & Iren, D. "Applying Scrum in Data Science Projects," *In 2020 IEEE 22nd Conference on Business Informatics (CBI),* Vol. 1, pp. 30-38. IEEE. June 2020.

[35] Qurashi, S. A., & Qureshi, M. (2014). "Scrum of scrums solution for large size teams using scrum methodology". *Life Science Journal-ACTA Zhengzhou University Overseas Edition, online May 2014,* Vol. 11, pp. 443-449. 2014.

[36] S. Dorairaj, J. Noble and P. Malik, "Understanding Lack of Trust in Distributed Agile Teams: A grounded theory study," *16th International Conference on Evaluation & Assessment in Software Engineering (EASE 2012),* pp. 81-90. 2012

[37] M. Paasivaara, C. Lassenius, D. Damian, P. Räty and A. Schröter, "Teaching students global software engineering skills using distributed Scrum," *2013 35th International Conference on Software Engineering (ICSE)*, pp. 1128-1137. 2013

[38] Beedle, M., Devos, M., Sharon, Y., Schwaber, K., & Sutherland, J. "SCRUM: An extension pattern language for hyperproductive software development," *Pattern languages of program design*, Vol. 4(1), pp. 637-651. 1999.

[39] Santos, N., Fernandes, J. M., Carvalho, M. S., Silva, P. V., Fernandes, F. A., Rebelo, M. P., ... & Machado, R. J. "Industrial interoperability issues when adopting Scrum in research projects: the case of the iFloW system," *2016 International Conference on Interoperability for Enterprise Systems and Applications*. April 2016.

[40] Hanslo, R., Mnkandla, E., & Vahed, A. "Factors that contribute significantly to Scrum adoption," *In 2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 813-821. IEEE. September 2019.

[41] Luz, M., Gazineu, D., & Teófilo, M. "Challenges on adopting scrum for distributed teams in home office environments," *World Academy of Science, Engineering and Technology,* Vol. 59, pp. 308-311. 2009.

[42] J. Linaker, S. M. Sulaman, M. Höst and R. M. de Mello, (2015) "Guidelines for conducting surveys in software engineering", [online], Available: https://portal.research.lu.se/portal/en/publications/guidelines-for-conducting-surveys-in-software-engineering(8ac54dbe-b7ac-4244-9c43-0f0d157efa26).html [August 15 2021]

APPENDIX A

SELECTED PAPERS IN DETAIL

List of the selected literatures in this study

Titles	Authors	Digital Library	Year
Distributed Scrum in research project management	Marchesi, Michele, Katiuscia Mannaro, Selene Uras, and Mario Locci.	Springer Link	2007
Challenges on adopting Scrum for distributed teams in home office environments.	Luz, Marlon, Daniel Gazineu, and Mauro Teófilo.	Google Scholar	2009
Scrum in research	Ota, Martin.	Springer Link	2010
Adapting Scrum to Managing a Research Group.	Hicks, Michael, and Jeffrey S. Foster.	Google Scholar	2010
On empirical research into scrum adoption.	Paulk, Mark C.	Google Scholar	2011
Adapting and using Scrum in a software research and development laboratory.	Lima, Igor Ribeiro, Tiago de Castro Freire, and Heitor Augustus Xavier Costa.	Google Scholar	2012
Issues and challenges in Scrum implementation.	Akif, Raza, and Hammad Majeed.	Google Scholar	2012
ARDev: a methodology based on scrum principles to support research management on software technologies.	Bezerra, Daniella Rodrigues, Arilo Claudio Dias-Neto, and Raimundo da Silva Barreto.	Google Scholar	2014
Problems in the adoption of agile-scrum methodologies:	López-Martínez, Janeth, Reyes Juárez-Ramírez,	IEEE	2016

A systematic literature review.	Carlos Huertas, Samantha Jiménez, and Cesar Guerra-García.		
Industrial interoperability issues when adopting Scrum in research projects: the case of the iFloW system.	Santos, Nuno, João M. Fernandes, M. Sameiro Carvalho, Pedro V. Silva, Fábio A. Fernandes, Márcio P. Rebelo, Diogo Barbosa, Paulo Maia, Marco Couto, and Ricardo J. Machado.	Research Gate	2016
Scrum adoption challenges detection model: SACDM.	Hanslo, Ridewaan, and Ernest Mnkandla.	IEEE	2018
Adapting the scrum framework for agile project management in science: case study of a distributed research initiative.	Hidalgo, Enric Senabre.	Google Scholar	2019
A Novel Software Development Methodology for Research-Based Software Projects.	İbrahim Cereci, & Ziya Karakaya.		2019
Factors that contribute significantly to Scrum adoption.	Hanslo, Ridewaan, Ernest Mnkandla, and Anwar Vahed.	IEEE	2019
Applying Scrum in Data Science Projects.	Baijens, Jeroen, Remko Helms, and Deniz Iren.	IEEE	2020

APPENDIX B

SEMI-STRUCTURED INTERVIEW QUESTIONS

My name is AHMED ALKHARAM I am a software engineering master's student at Atilim University in Ankara, Turkey. The purpose of this Interview is to help fulfill this research study to achieve its purpose which is modifying Scrum for research-based software development projects. Your decision to participate or decline participation in this study is completely voluntary and you have the right to withdraw your participation at any time. All comments and responses are anonymous and will be treated confidentially. For more information, you can contact me at Alkharam.aasalem@student.atilim.edu.tr or my supervisor Professor Ziya Karakaya on his email ziya.karakaya@atilim.edu.tr.

Goal of this interview:

This study aims to identify the possible issues/challenges when using Scrum in research-based projects and to address how to overcome those challenges/issues that occurred when using Scrum in research-based projects.

Glossary

Research-based project: "Any creative systematic activity is undertaken in order to increase the stock of knowledge, including knowledge of man, culture and society, and the use of this knowledge to devise new applications." UNESCO Statistical Yearbook, UNESCO, Paris, 68 and 65, Chap. 5.

Semi-structured Interview Questions

Questions related with the team members:

- **What best describes your role in the project? "The project refers to the Scrum or agile-based research project you participated in most recently"**
- **How many projects have you been involved in using Scrum in research-based software development projects?**

1. **Have you and your team members had experiences with Scrum?** (NO) Did your company/organization train you on using Scrum? (NO) How did this issue "Lack of training/experience" affect your project?

2. **Did your team follow Scrum characteristics in all stages of the research project?** (YES) To what percentage did you follow Scrum in the project?

Questions related to the problems:

3. **Was there any type of collaboration challenges between team members, or between team members and stockholders?** What do you think were the reasons? And how did you overcome this issue?

4. **What type of communication issues did you face between your team members or between team members and stockholders? what was the reason for it?** What type of solution did your team do to overcome the communication challenges?

5. **Did you face a lack of trust among your team members?** (YES) How did this issue affect your research-based project? And how did you overcome this issue?

6. **Was the limitation in Scrum team members an issue?** (YES) How did it affect the project? And how did you overcome this issue?