TU Delft

Scrum for Hardware Development Projects

A study to fit the Scrum method in Hardware Development Projects

Satish Jyotsni | 4916476

MSc Construction Management and Engineering

August 2020

Image source: Wilson, F (2020). (https://www.ntaskmanager.com/blog/best-practices-for-resource-management/)

Colophon

Research title: **Scrum for Hardware Development Projects**
A study to fit the Scrum method in hardware development projects

Date: **August 25, 2020**

Writer: **Satish Jyotsni**

Student number: **4916476**

University: **Delft University of Technology**

Faculty: **Civil Engineering and Geoscience**

Masters study: **Construction Management and Engineering (CME)**

Graduation Committee

Committee Chair: **Prof. Dr Hans Bakker**
Professor & Chair, Management of Engineering Projects
Section of Integral Design and Management, Faculty CEG, TU Delft

First Supervisor: **Dr Ir. Marian Bosch-Rekveldt**
Assistant Professor
Section of Integral Design and Management, Faculty CEG, TU Delft

Second Supervisor: **Ir. Bart Bluemink, R.G.H**
Lecturer and PhD candidate
Section of Management and Organization, Faculty IDE, TU Delft

Company supervisor: **Ir. Karin Rosch**
Manager project management

Preface

With the completion of this thesis, I would finally have my master's degree in Construction Management and Engineering from the Delft University of Technology. I consider this as my most significant achievement to date. This report is the outcome of a challenging eight-month of graduation work. I must say it was a very insightful learning experience, which allowed me to combine practical knowledge with the theory that I have learned.

This thesis would not have shaped without the guidance of my graduation committee. I would firstly like to thank my first supervisor Marian Bosch, because of whom I first got to know about this topic/opportunity at a company. She was always available, gave constructive advice and guided me towards the right path. Next, I would like to thank my second supervisor Bart Bluemink for giving all the support and feedback throughout the project. A Special thanks to my chair Hans Bakker, for a thorough assessment of my report and clear feedbacks in all my milestones, which took my work to the next level.

Secondly, I would like to thank my company supervisor Karin Rosch, for giving me this opportunity and believing in me that I can do this. Even during the quarantine period, she always motivated and encouraged me. Big thanks to all the project leaders and interviewees for taking time out of their busy schedules and helping me with the case studies.

Finally, none of this is possible without my parents Mr Satish and Mrs Madhu; because of them, my dream has come true, and I am forever grateful to them. Then I want to thank my brother Prasana Aditya and sister Harshini for all the constant help throughout all stages of my life. My friends are my biggest support system, and I can proudly say I got the best ones. I would like to thank Ayush for cultivating this vision of studying abroad and for always being there. Then I want to thank the pillar of this masters journey Ishani, Rian, Ankita, Sangeeta, Sruche, and Shiyam, for continually believing in me. They showed trust in me during my tough times, made this journey bearable and never let me lose hope. I am also thankful to my amazing friends in delft; Khyathi, Nikhil, Atul, Yash, Madhura, Abhishek and Akshay for constantly being there to talk and support me throughout this journey. With all of this support and encouragement, I feel super happy and excited to say that I have finally reached my end goal, and I am ready for future endeavours.

Satish Jyotsni
August 2020

Executive Summary

The trends in the high tech sector push towards creating and delivering better products to cope with the increasing demands of the customers. The projects of the high tech sector are referred to as Hardware (HW) development projects, which consist of both hardware and software (SW) development. Many organisations have been using traditional project management approaches for years for such projects. However, for the past few years, the organisations are considering or in the progress of shifting towards the agile project management approach. As agile management has potential benefits, which includes high involvement and satisfaction of customers, a better quality of deliverables, and adapt to changing requirements in a project. However, it is possible to integrate practices of the agile method into the traditional project management approach, and when combined, this would improve the functionality of the product less influencing the cost and time. Therefore mixing both the traditional and agile approaches can have a significant effect on the performance of the project.

A common framework of the agile project management approach is the Scrum method, commonly used for software development. The Scrum method is made up of Scrum roles, ceremonies, and various Scrum practices. The practices of Scrum are referred to as the elements of Scrum. Studies show that the Scrum method can also be used for hardware development. However, to get full advantage of the scrum method for hardware development, it needs to be tailored to the needs and type of the project.

The research aims to explore the application of Scrum for hardware development projects. The objective is to formulate suggestions on elements of Scrum that can be applied in the management of hardware development for the benefit of the projects.

Research question

The main research question that this research aims to answer is:

How can the traditional project management approach applied for HW development projects be improved by using the scrum method?

This question is answered by a combination of literature study and multiple case studies of the projects in the company. The literature study extensively discusses the characteristics of hardware development, the coexisting of HW and SW in complex products, and the process of HW development. In preparation for the case study, a literature study was performed to review the traditional project management approach and the scrum method. Based on the literature study, a theoretical framework was developed, describing scrum practices and their benefits. The benefits known from the theoretical framework will be used as background knowledge to investigate the fit of Scrum elements in the current management approach. Further, the applicability of scrum practices for HW development is decided based on the agile value and principle; each Scrum practice is related.

Explorative and descriptive case study

In the case study, the goal was to explore and describe the various projects of the company and understand their management practices. For this purpose, three projects were selected for investigation. The company, in general, uses a waterfall approach for planning HW development and scrum method for planning SW development. Therefore, three members of different specialisations

were interviewed per case (HW development, SW development, and the project leader). Since the HW development projects are quite complex, it was useful to find the complexities that influence the most. To assess the complexities in the project, a complexity (TOE) framework developed by Bosch-Rekveldt (2011) is used to determine and to get complete awareness of complexities in HW development projects. Based on the project documents, complexity assessment, and interview results, case analysis was performed describing the management approach used on various themes of the project. With this, the management approach used in practice was understood. For the cross-case analysis, the cases were compared on different subjects (project characteristics, project complexity, and project management approach).

The analysis of cases showed some common problems currently faced in the management approach of HW development projects. A list is made based on the complexity elements, and the problems identified by the case study and complexity assessment. Based on the elements in this list, suggestions on the most promising Scrum elements to manage each element is made. However for some elements like: "Lack of customer involvement," "Lack of resources and skill availability," "Lack of Experience with parties involved," and "External risks" no Scrum elements could be suggested.

Expert meeting

An expert meeting with three professionals with over 15 years of management experiences was organised individually. This validation meeting aimed to discuss all the suggestions of scrum practices and their applicability in managing the complexity elements and problems. From the results from the expert meeting, it is seen that experts are not in favour of the suggest scrum practices for seven out of seventeen elements, as they are not convinced that it will help. Based on the conclusions from expert meeting, Scrum practices such as Sprints, scrum board, burndown charts, daily stand-ups, and Sprint reviews/retrospective were considered for HW development projects.

Conclusion

The main aim of blending both approaches is because some aspects from both approaches achieve better results than others. The literature study shows that Scrum can fit (some practices can be applied directly, and some needs to be adapted) for HW development projects. The results of the interviews show that the company is skilled in doing HW development projects with the traditional (waterfall) project management approach. The data gathered from the interviews and complexity assessments illustrate that large HW development project follows a traditional project management approach irrespective of their complexity. Also, it shows that the differences and number of complexities did not play a role in choosing the management approach. It also shows that the customer has a substantial impact on the way of working.

This research determines the characteristics of the HW development project and the difference between the traditional project management approach and agile project management approach. Also presents the suggestion of scrum practice after considering the inputs from the experts meetings that can be applied in managing specific complexity/problem. The scrum practices that are considered for adoption in HW development projects are *Sprints, Scrum Board, Burndown Charts*, *Daily stand-ups,* and *Sprint reviews/retrospective.* The suggested scrum practices are beneficial for the following elements:

- *Uncertainties in scope*
- *High number of project goals*
- *High number of tasks*

- *Dependencies between tasks*
- *Involvement of different technical disciplines*
- *Interfaces between different disciplines*
- *Strict quality requirements*
- *Number of locations*
- *Involvement of different time zones*
- *Number of different nationalities*

Therefore, the suggested Scrum practices, when followed, will improve the traditional project management of HW development projects in managing the specific complexity element/problems. The suggested Scrum practices adds a value of increased productivity, visualisation, and effective communication, which influences the overall collaboration with the team and customer in the project. However, Scrum elements can fit for HW development projects, but to implement scrum practices, a sense of urgency and cultural change is needed in the company to apply, and this will take time.

Further, to implement the Scrum practices at the company, four leading scrum practices - Daily stand-up meetings, time-boxed sprints, Product backlog, and Sprint reviews/retrospectives are considered. An adoption procedure of scrum practices is proposed to make it actionable in the company's way of working.

Research recommendations

- The research recommends the organisations to discuss the meaning of agile and practices of Scrum with the project team, so all of them have the same mind-set before implementing.
- The results of the complexity assessment could be considered as a starting point to determine specific project characteristics. With this, the right competencies for the team can be chosen based on the project characteristics.
- Inputs from experts scrum for HW development should be considered in the pilot project for better understanding of the concept and workload.
- A full implementation plan should be made depending upon the needs of the project and should be given to all the members when following a pilot project
- Moreover, changes are inevitable in complex projects; consequently, they should be embraced rather restricted; using agile practices gives flexibility and increase satisfaction for the customers.

Table of Contents

Abbreviations

HW –Hardware

SW- Software

PGP – Product Generation Process

SOW – Statement of Work

PMP- Project Management Plan

WOW –Way of Working

List of Figures

List of Tables

1. Introduction

This chapter introduces the research. Section 1.1 gives the background about projects in the high tech sector and a brief introduction to a traditional project management approach and agile project management approach. Section 1.2 provides the general information of the organization where the research is undertaken. Section 1.3 formulates the problem statement of the research. Finally, section 1.4 elaborates on the outline and structure of the report.

1.1 Background

The government of the Netherlands classifies the high tech sector as one of the top nine sectors. The high tech sector acts as a facilitator to other sectors and provides realistic solutions to the challenges of technology (High-Tech Industry in the Netherlands, 2016). The sector interlinks with manufacturing industries and has overlapping technologies from system architecture to production. The hardware development projects of the high tech sector have end deliverables made of both hardware and software development. With the increasing demand, the project of the high tech industry requires development projects from a new family of products or a new product completely based on advanced technology (Shenhar, 1993). According to Shenhar (1993), planning is essential in managing these kinds of projects. Considering the amount of increasing uncertainty and risk, it is necessary to allow for crucial trade-offs. The hardware development projects involve a considerable amount of risks due to the involvement of high-paced technology. It also requires project managers to be flexible enough to take advantage of the existing technology and not overrun the time and cost (Shenhar, 1993). Therefore, this kind of project requires a management approach to be flexible and open to embracing technological development, but also to expect changes along the way.

HW development projects are managed by the traditional project management approach (Lima et al., 2015, Drechsler, & Breiter, 2007) However, over the past few years, an increase in complexities and uncertainties have made the projects more dynamic (Sakal, 2005). A common problem in the project management world is the failure of projects to remain within the budget and time (PMI, 2018). The traditional project management approach only fits for tasks with linear relationships and does not consider the complexities and dynamics of modern projects (Špundak, 2014). The organizations face difficulties in coping with the rising complexities, and since every project is different, therefore, one size and type of project management approach does not fit all.

A significant development in project management is the introduction of the agile project management approach. According to Punka (2012), Agile was created to address drawbacks in traditional methods. Though at first, the Agile project management approach was applicable only for software development projects, it recently has found its way in hardware development. Despite the differences between hardware and software development, the value of agile is still realized (Hanser, 2010). The implementation of agile methods in development projects, according to Reynisdottir (2013), improves communication, reduces redundant work, and gives a clear image of the project to the team. Further, the study of Karlstrom & Runeson (2006) shows that the agile method increases the quality of the deliverables. There is also a higher chance of success in a project when agile methods are used (Serrador & Pinto, 2015).

According to Karlstrom & Runeson (2005), it is feasible to integrate agile practices in the traditional project management approach, and when combined, this would improve the functionality of the

product less influencing the cost and time. Therefore mixing both the traditional and agile approaches can have a significant effect on the performance of the project (Schuh et al., 2016). The selection of the right elements from both management approaches is needed for the success of the project.

1.2 About the company

'The company' refers to the company from which the research in this thesis originated. The company performs projects for specific clients, referred to as 'the customer.' The company combines the strength of a multinational with a flat and open organization. The company has an informal working atmosphere, which gives priority to advancement opportunities and long-term ambitions. They innovate through a combination of craftsmanship, entrepreneurship, and the use of high-quality machines. They are also a global player in the field of complex and innovative mechatronic systems. They develop and produce components and modules as well as complete systems for high-tech production equipment, analysis systems, and medical systems. It is a tier-one contract manufacturing partner, which means they make products based on the requirements of the clients. It is a worldwide supplier of modules, mechanical components, and mechatronic systems for various industries.

Technology and development (T&D) of the company is responsible for the development of new products. It is a matrix organization with a large development group. There are two types of development projects; built-to-specification and installed base projects. The projects are based on customer demands only, and no own products are developed. The project management way of working for the hardware development is based on the traditional (waterfall) approach, while for software development agile approach (Scrum method) is applied.

The company uses the Project Management Plan (PMP), which is the main communication document and uses it as a reference for all decisions made on the project and for clarification of unclear ideas. It is to ensure consistent management of the project and verify the customer expectation. It is available to all the project members, including the customers. The development process (product generation process) followed in the company is used for introducing new products for the customers. Product Generation Process (PGP) consists of formal control points, with uniform milestones and deliverables, and has clear assigned roles and responsibilities. Figure 1 represents the PGP of the company.

Figure 1 Product generation process (based on the company documents)

The company follows a project management process, which is the way of working for the management of their projects, and it is the process followed by the project leaders. Figure 2 represents the project management process combined with the product generation process.

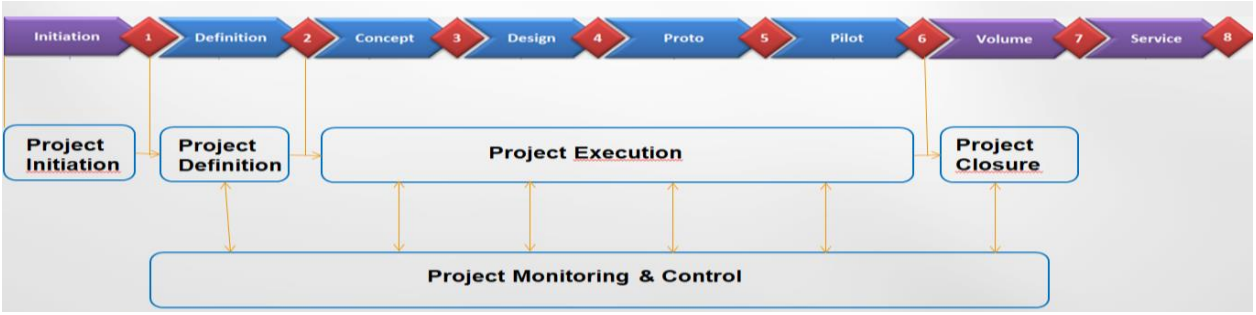


Figure 2 Project management process (based on company documents)

For exploratory purposes, initial discussion meetings were held with three project managers individually to get a primary understanding of the development projects. The discussions revealed that Built-to-Specification is a project in which a new solution is developed for the clients until the release for volume production. Installed-Base projects consist of small changes or improvements to products after they are installed at the customer site. Based on the discussions, it could be understood that the projects in the company are complex because of the existence of volatile requirements, sometimes are a first of kind project, consist of international culture and pressure to follow the fixed budget of the project. The representatives from the organization were very much interested in applying an agile project management approach to their hardware development projects. However, they were not sure how this might coexist with their current management method.

1.3 Problem statement

The company works in the high tech sector, which executes projects consisting of both hardware and software development. Several factors play a role in determining the right management approach that is most appropriate for such a project. The company currently uses a traditional project management approach for hardware development. However, the study by Bianchi et al. (2020) shows that the mixing of approaches can outperform any method applied purely. The study of Boehm & Turner (2003) in Bianchi et al. (2020) suggests that there is a need to balance both the traditional and agile approaches. However, the debate exists on how to mix both methods and balance them. The representatives of the company are interested in implementing the Scrum framework of the agile approach for hardware development in their projects, driven by the opinion that Scrum is the most popular agile management method (Schwaber & Beedle, 2002). The focus of this research is, therefore, on applying the Scrum method for hardware development in their projects. The hardware development projects are bound to have many complexities. For managing the complexities, it is necessary to know the complexities that play a significant role in such projects, and then based on the complexity elements, Scrum practices could be selected (Sohi et al., 2016). The study from Weinreich et al. (2015) concludes that most of Scrum elements can be applied directly to the hardware development project while some of them need customization. However, it is still unclear as to what elements of the Scrum method can be applied for hardware development and during which phase of the project.

1.4 Report structure

Figure 3 represents the structure of the report. Following this chapter of the introduction, chapter 2 is about the research design, where the researcher formulates the research question and research methodology. Chapter 3 is the literature study and ends with a theoretical framework. Chapter 4 is about the case study; it elaborates on the selected cases and the analysis of all data gathered from the documents, semi-structured interviews, and complexity assessment. Chapter 5 is about the cross-case analysis and discusses the suggestions based on the investigations. Chapter 6 discusses the practicality of the suggestions based on the meeting with the experts. Chapter 7 discusses the outcome of the research and reflects on the findings from the research. Finally, chapter 8 concludes the research and answers the main research question, followed by recommendations for further research and practice.

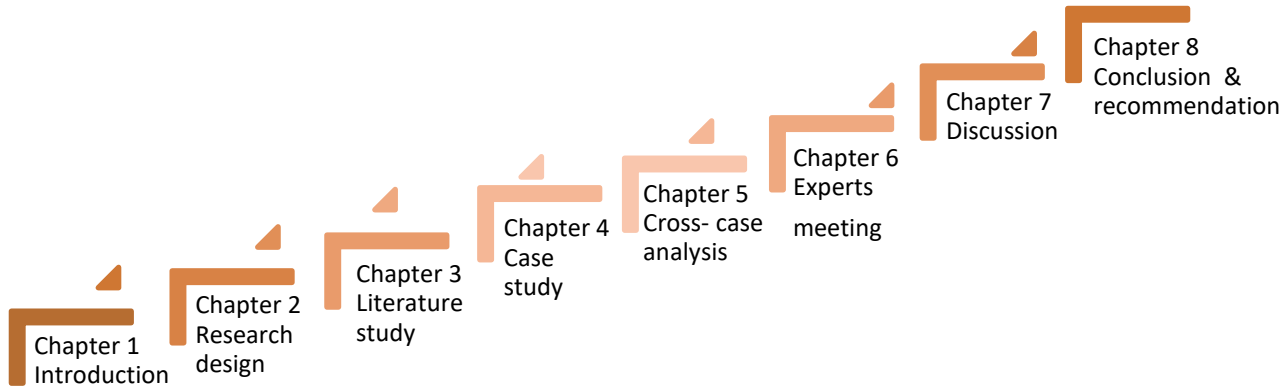


Figure 3 Structure of the report

2 Research design

This chapter formulates the research design. Section 2.1 presents the research goal and objective of the thesis. Section 2.2 formulates the main research question based on the objective of the research and states the sub research questions. Section 2.3 defines the scope of the research. Finally, section 2.4 explains the methodology that is adapted for this thesis.

2.1 Research goal and objective

The company wants to get deeper insights on managing hardware development projects and suggestions on agile project management approach for HW development. The research aims to explore the application of Scrum for hardware development projects. The objective of this research is to make suggestions on which elements of Scrum can be applied in the management of hardware development for the benefit of the projects.

2.2 Research question

To fulfil the objective of the research, the researcher framed the Main Research Question (MRQ):

MRQ. How can the traditional project management approach of HW development projects be improved by applying elements of scrum?

To answer the MRQ, the following Sub Research Questions (SRQ) was formulated.

SRQ1. What are HW development projects?

SRQ2. What is the Traditional project management approach used for hardware development projects?

SRQ3. What is the Scrum method and what are the practices of Scrum that can be applied for hardware development projects?

SRQ4. How is the traditional project management applied for HW development projects in practice?

SRQ5. What practices of Scrum can be adopted for the benefit of HW development projects?

2.3 Research scope

The focus of this thesis is about finding the elements of Scrum that can be applied to hardware development projects. It is out of scope to define an entirely new methodology or in-depth change process, but instead give suggestions on the use of Scrum elements within the context of introducing agile practices to the current management approach. The research focuses on two different types of projects in the company, which includes Built-to-Specification and Installed-Base projects, as these are the two main types of projects done at the company.

2.4 Research methodology

According to Thornhill et al. (2009), research is classified into four different kinds of studies. The type of study chosen for this research is the combination of explorative and descriptive. The explorative

aspect of the study gives new insights into understanding the nature of the problem. The descriptive study is more observational and gives a clear assessment of a situation or an event. This study aims to explore the undertaking of Hardware development projects in the company and analyze the complexities encountered. The Case study approach is chosen to describe the projects of the company, as it will give an outlook on understanding the projects thoroughly (Yin, 2014).

A literature study is done on the topics of characteristics of HW development and the traditional project management approach used for HW development. Based on the relevant works of literature, the researcher formulates the characteristics of hardware development. Further, the literature on the Traditional project management approach and Agile project management approach is investigated. Next, the literature study on the Scrum framework, its practices, and its application on hardware development is performed. The literature study phase concludes with a theoretical framework about the benefits of Scrum practices. The first three sub research questions are formed to gain a complete understanding and the management of hardware development projects.

The fourth sub research question aims to understand the current management approach of hardware development projects while focusing on the project characteristics and project complexities in the company. As the nature of the research is exploratory, the case study approach is selected. The case study setup and the selection of cases and respondents are established. The primary data collection method was through face-to-face semi-structured interviews with the selected participants of the case. The semi-structured nature of interviews is chosen because it allows the interviewees to express their experiences more openly, and the interviewer has the opportunity to ask to follow up questions. However, before the interview, the documents related to the cases are studied, and a complexity assessment is done by the respondents of the interview. The interview questions were formulated based on the theoretical framework derived from the literature study. All the data gathered from the complexity assessment and interviews are sent to the respective participants for validating.

The data collected from three cases based on the documents, results of complexity assessment, and nine interviews (three per case) are analyzed. Qualitative data analysis is performed by describing the answers to the interview on various themes. From the complexity assessment, the element that is marked by two or more is selected and discussed. The data is analyzed for each case. Finally, the last sub research question is formed to find the applicability of scrum practice for the benefit of HW development projects. Therefore, the cross-case analysis is performed, and the comparison is drawn between project characteristics, complexity elements, and problems identified in the cases. Based on the literature study of scrum practices on HW development, suggestions of Scrum practices are formulated for the complexity elements and problems identified. Finally, the experts from the company further reflect on the applicability of the suggested scrum practices, which is taken into account to formulate the final suggestions. The research is concluded, and final remarks on the adoption procedure for the suggested scrum practices are made based on further discussions. This phase is concluded by discussing the recommendations for practice and future research. Figure 4 represents the research methodology as a flow chart consisting of significant steps and shows how the sub research questions are answered throughout the process.

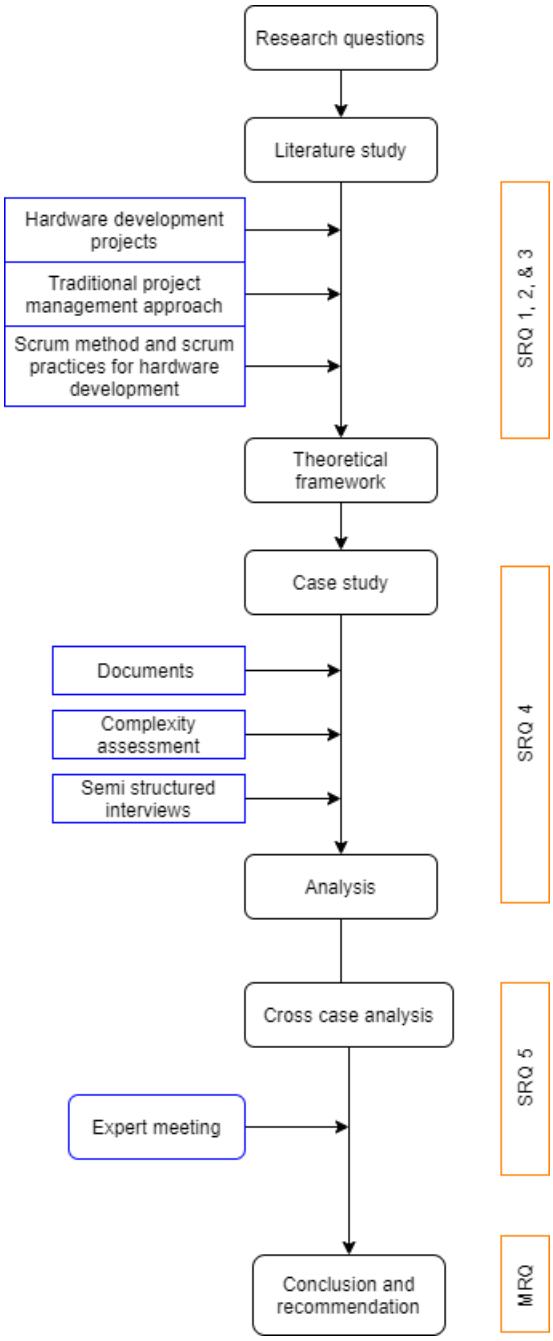


Figure 4 Research methodology

3 Literature study

This chapter gives an overview of the research topics to answer the sub research questions 1, 2, and 3. It starts with the discussions on the nature of hardware development, its characteristics, the coexisting hardware (HW) and software (SW) in complex products, and lastly about the process of HW development. Following that, the traditional and agile project management approaches are discussed and finally explaining Scrum and its practices. The chapter concludes with the theoretical framework on the benefits of the Scrum and its elements.

3.1 Hardware development projects

The hardware refers to tangible products. The term Hardware development in this research refers to the development of HW from the design phase until it reaches the manufacturing stage without the details about manufacturing. However, the HW development projects also involve SW development, which refers to a computer program that acts as a medium for interaction between the customer and the HW (Permana, 2015). The goal of this section is to discuss and understand HW development as it is the focus of the research. Section 3.1.1 explains the nature of HW development, followed by section 3.1.2, which explains the coexisting of HW and SW. Lastly, section 3.1.3 discusses in detail the process of HW development.

3.1.1 Nature of hardware development

The development processes of hardware and software have similar sub-phases (Drechsler & Breiter, 2007). Brandl et al. (2018) state that HW products consist of large physical components that cannot be refactored after manufacturing. Also, states that HW designs are constrained with the need to incorporate standard parts. In both cases of HW and SW development, coding is used to simplify the development process (Drechsler & Breiter, 2007). On the other hand, Brandl et al. (2018) also point out the differences in HW development compared to SW development is that they are less malleable, higher cost for changes, less scope for qualitative changes compared to quantitative, more lead-time, more upfront design cost, and fewer tests for validation due to involvement of expensive equipment.

The difference between hardware and software, according to King et al. (2015), is that the computational model is different for both. The significant difference between hardware and software, according to Augustin & Schabacker (2019), is that hardware development requires physical prototypes. Adding to differences in HW development, the study of Schuh et al. (2016) mentions that HW is not easy to change, has more upfront architectural work, HW is usually designed, and tested to work for a longer period. The costs of the HW increase at the end of the development cycle, and the changes during the development cycle disrupt HW development. Similarly, the study of Fuchs & Golehhofen (2019) points out more differences in HW development. According to the study, HW cannot be redesigned as quickly as in SW, and the physical requirements and constraints are critical for HW. Also, the prototype for verification and validation requires more workings steps and resources. While developing HW, there is a higher complexity in communication within the development teams. The study also mentions that HW requires long procurement and production steps, has a high dependency on the suppliers, and requires more testing equipment.

The discussion mentioned above analyses the differences in HW development when compared to SW development and therefore helps in framing the overall characteristics of HW development. Table 3.1 represents the characteristics of HW development.

Table 3.1 Characteristics of HW development compared to SW development (own illustration)

HW development	Less malleable
	Higher cost of change
	Change in the middle of development cycle causes disruption
	Focuses more on quantitative than on qualitatve improvements
	More upfront architectural work
	Requires expensive test facilities
	Developed for longer ageing period
	Cost increases during the development cycle
	More time taken for redesigning
	Critical physical requirements and constraints
	High complexity in communication within the development teams
	More workings steps and resources required for the prototype in verification and validation
	Long procurement and production step
	High supplier dependency
	High lead time
	Similar sub phases of developement process

3.1.2 Coexisting of hardware and software

The complex product/system produced has not only hardware but also software (Fuchs & Golehhofen, 2019). The cost and complexity of a hardware system can be reduced with the help of software and vice- versa. The SW optimizes the product according to the needs of the customer by helping to meet the change in requirements. The connection between hardware and software is considered very important (King et al., 2015). For the software to be embedded in hardware, it should combine various perspectives from the conception to development. It is called Co-design, when there is an existence of a combination with various software, electrical, and mechanical design (Lima et al., 2015). The customer demands integrated complex solutions, which increase the interfaces and integration between HW and SW (Fuchs & Golehhofen, 2019). The increase in digitalization and automation has created constant innovation of the new products and concepts that make the differences between HW and SW likely to change and making it more adaptable.

3.1.3 Hardware development process

Since the focus of the research is on hardware development, it is essential to understand the process of development in HW. According to Drechsler & Breiter (2007), the overall workflow in the hardware development process goes from *Initialization* to *Re-use*. The development stages in hardware are interconnected and dependent, and if there is an error in any phase, the process goes back to the previous phase and corrects it from the start (Lima et al., 2015). The process of product development is always subject to both technological and market uncertainty (Albers et al., 2017). In

hardware design projects, most of the problems existing in the verification stage are not real design problems, whereas they are problems caused due to incomplete design specifications (Bentley, 2001 in Drechsler and Breiter, 2007). According to Drechsler & Breiter (2007), the following are some specific problems that can occur and therefore require special attention in the HW development process:

- Addressing new requirements after the HW is developed leads to the difficulty in managing risks.
- Designers turned project managers cause non-clarity of roles in the development teams. Since they have technically strong knowledge about the product, there is a high possibility that they will influence the designers in the team.
- The time invested in specifying the details of the HW to build in the early stages of the project is not enough. This makes unclear specifications; the leading reason for the projects to delay or fail.
- HW designers prefer to work individually, rather than working in a group.
- Rapid changes in technology make the planning of resources nearly an impossible task.
- Difficulty in specifying the status of the project as testing of many systems cannot happen until the final design is ready.

3.2 Traditional project management

A project is a complete process from the initial phase of planning to the final phase of delivery of the product (Drechsler & Breiter, 2007). A project is undertaken to create a unique product or service. It has a definite beginning and end; therefore, it is a temporary endeavour. According to Kerzner (2017), project management is the application of knowledge, tools, and skills to achieve project requirements. Brinkkemper (1996) defines this traditional project management approach as a structured way with a set of rules and guidelines to manage the projects. Charvat (2003) defines the same as a set of guidelines and principles that are tailored for a specific situation or a project. The traditional project management approach has a basic idea of the projects to be simple with clear boundaries, and predictable (Spundak, 2014). This is because generally, the development of projects advances linearly, and the main goal in the management of projects is to follow a plan and complete it in a defined time, scope, and cost. According to Sutherland and Ahmad (2011), the following are the main characteristics of the traditional project management approach and are represented in Figure 5.

- It is divided into different homogeneous activities where it transfers from one to another phase, following a series from requirements, design, implementation, verification, and maintenance.
- It is sequential because one phase starts when the previous phase is completed.
- It does not repeat without a formal procedure. In the case of redo, all the phases are executed again sequentially, which makes it non-iterative.
- It follows an order in which all the activities and phases will be executed; therefore, it is driven by a detailed plan. Each phase has various activities to be fulfilled.

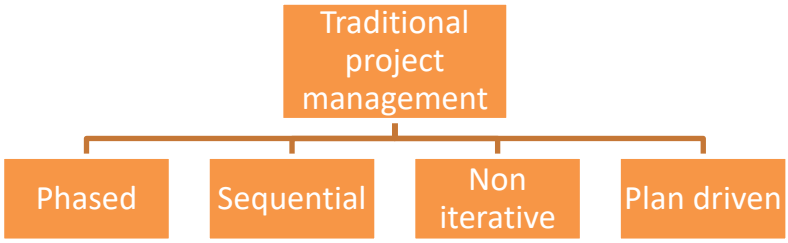


Figure 5 Characteristics of traditional project management (Sutherland and Ahmad, 2011)

The methodology of project management has defined methods, techniques, procedures, and the best practices applied to a project (Jugdev et al., 2001). Originally the practice was developed by W. Royce, and Dr. Winston, where the overall planning is done before the development starts and testing is done after the development phase is finished (Royce 1970). This kind of development process was known as the waterfall development model and is represented in Figure 6. It is called the waterfall model because the structure of the model looks like a waterfall, where every phase starts after the completion of the previous phase.

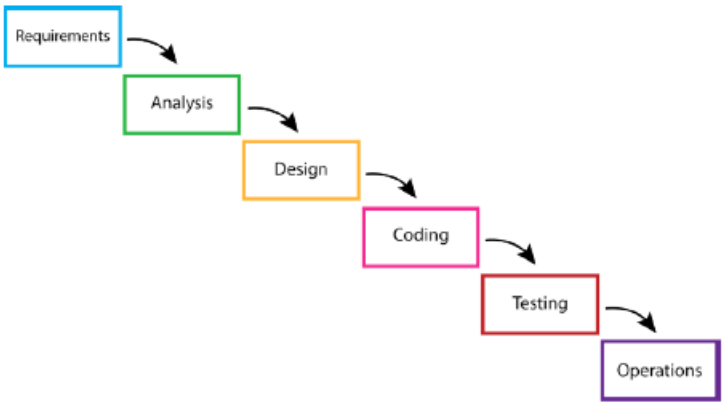


Figure 6 Waterfall model (based on Royce, 1970)

Robert G. Cooper developed the stage-gate model with linear development stages in the 1980s. It is a better version of the waterfall model to improve the effectiveness and quality of the product generation process. A stage-gate model describes a work process from the idea to the delivered product (Karlstrom and Runeson, 2005). The model is a generic representation of different stages of product development. More comprehensively, the model is connected to a technical process model like the waterfall (Cooper, 2001 in Karlstrom & Runeson, 2005). Each stage in the model is connected with a gate, which is the decision point for the management and the sponsors of the project to make decisions about whether or not to proceed to the next stage. Figure 7 shows the stage-gate model of five stages (Cooper, 2001). It typically consists of 4 to 7 stages depending on the size and complexity of the project.

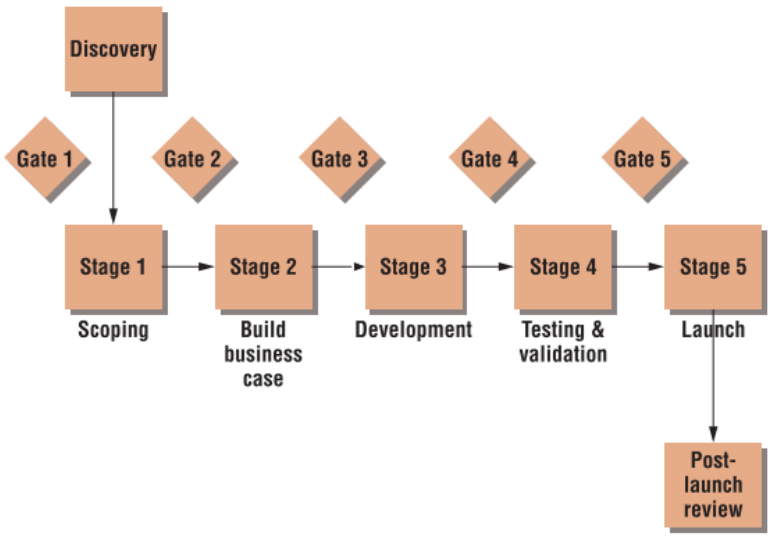


Figure 7 Stage-gate model (Cooper, 2001)

According to Toljaga-Nikolic (2017), the other models of traditional project management approaches are process-oriented (plan-driven) project management, which is presented by PMI (in PMBOK) and PRINCE2. The process-oriented project management consists of defined concepts related to the traditional approach. It is made of five groups of process, which is the basis for managing the projects and has ten knowledge areas, which are characteristic of all projects and further help in building the field. PRINCE2 consists of frameworks, which divide the project into manageable and controllable stages. It consists of seven principles related to the process and areas that make a firm structure and controls the project.

Traditional development models are based on the following ideas (Punkka, 2012)

- Full system requirements are known at the start of the project
- Possibility to make a full detailed plan
- Possibility of the plan to be executed fully

Traditional development methodologies rely on predefined models to capture the whole progress of a project in advance. Moreover, the planning is usually based on work breakdown structure made up of work packages and milestones (Overhage et al., 2011). According to Hornstein (2015), Project management is successful when the project is completed or achieved within the allocated cost, time, and within expected technology and performance. Additionally, a project should be efficient and effective within prescribed resources and accepted under the requirements of the client. The success is, therefore associated with the ability to deliver with a necessary trade-off between scope, time, and cost. The study of Drechsler & Breiter (2007) suggests that the essential aspects of the project should be considered for a successful hardware development when using a traditional project management approach and is shown in Figure 8. For the success of the project, it is crucial to understand the technical, organizational, and social aspects of the project. In the overall success of hardware development, the project organization, and the management plays an important role (Hornstein, 2015). Besides, the study of Toljaga-Nikolic (2017) finds that the project characteristics act as the basis for selecting the management approach for the project. Moreover, when more than one approach is used, the criterion for the selection of management approach also depends on the efficiency and practicality of the project.

Also, to design an optimized product, it is vital to consider and cope with both external and internal changes. However, when hardware and software coexist in a product, then there exist strong dependencies, and such projects are more complex, and the trade-off decisions are harder to make. Many authors like Baccarini (1996), Williams (2005) consider that interfaces lead to complexity, and the management of projects with complexities requires considering various conditions that give rise to these complexities. According to Baccarini (1996), in Sohi et al. (2016), the complexity of the project is one such critical dimension in the project, which determines the type of managerial action required to complete the project. Therefore, it is important to understand complexities, as this would help in identifying the causes of problems and further help in improving the projects (Haas, 2009).

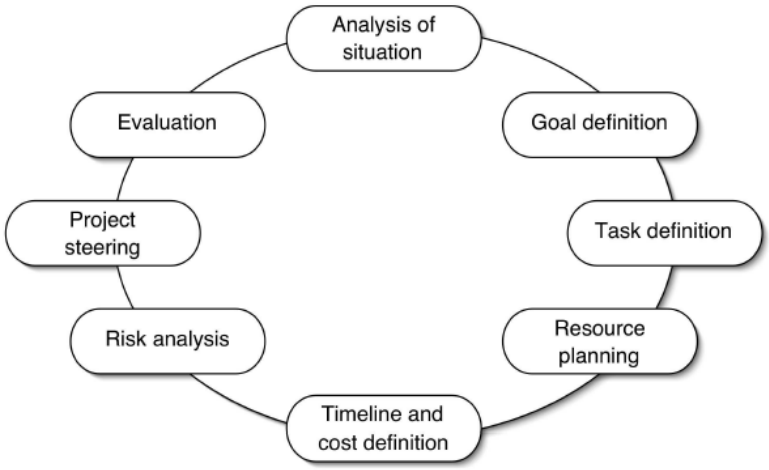


Figure 8 Important tasks in project management (Drechsler & Breiter, 2007)

The traditional (plan-driven) approach is effective in a relatively predictable and stable environment. However, many studies question the value of the stage-gate model in today's increasingly uncertain and fast-paced environment due to problems of excessive rigidity, budget overruns, delays (Cooper, 2014). It is required to take necessary actions in the project setting to manage the complexities that would improve the decision-making process and management style (Antoniadis et al., 2011). According to Geraldi (2008), projects demand order and awareness of the complexity and uncertainty. Traditional project management reflects order, but for the awareness of complexity, there is a need to assess the complexity. It helps in determining the project management approach or practice to manage it. The robustness of the traditional approach is considered as a critical disadvantage of this approach (Špundak, 2014). The use of a sequential process with predictive planning is less adaptive and non-flexible. Therefore this questions the idea behind the traditional management models. Punkka (2012) suggests that such development projects are best managed with the process of continuous integration and refined design, which is fulfilled by agile methods.

3.3 Agile project management

This section introduces agile project management. Section 3.3.1 defines the agile concept and its management approach. Section 3.3.2 discusses the differences between agile and traditional project management.

3.3.1 The concept of Agile

A new type of management approach has emerged for all the industries to cope with growing innovations and to address the drawbacks of the traditional project management approach (Williams, 2005). Organizations need to be able to respond to frequent and sudden changes in the market while still focusing on cost and quality. A team of 17 software developers came together, discussed, and compared various development methods and published the agile manifesto, also called the manifesto for software development (Dingsøyr et al., 2010). The word agile is used to describe the process model, and it means to be lightweight, alert, fast, and free moving (Permana, 2015). This approach is based on the iterative and incremental development principles, which have existed since the 1950s. Agile is a mindset supported by four values and twelve principles (mentioned in appendix A) and is attained through various practices. It is a way of thinking about overall activities in a project (Fuchs & Golehhofen, 2019). The main aim is to induce a constant flow in delivering the output and aid the organization to promote teamwork and reach the common goals faster, which tends to keep the customer happier. The main goal is to develop products rapidly and iteratively with the customer taking a significant part from the start of the project (Garzaniti et al., 2019). Thus, the customer can give direct feedback on the progress of development. The whole concept of agile is illustrated in Figure 9.

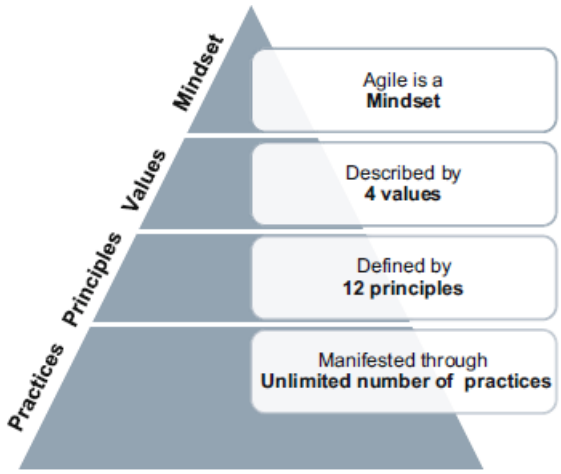


Figure 9 The agile mind-set (Sidkey (2015) in Fuchs & Golehhofen, 2019)

The iterative approach is also called incremental or spiral. With every successful release of the product at the end of the iteration, it eventually evolves into a complete product (Fuchs & Golehhofen, 2019). This approach is considered a good way to open up unknowns and also help in adapting to changing requirements. The focus is to create value in all stages of the product by having small deliveries in every phase.

3.3.2 Agile vs traditional project management approach

The thesis work of Hendriks (2019) shows that the study of Highsmith (2009) introduces agile as the replacement for the traditional iron triangle made of scope, time, and cost as the constraints of the project. The agile triangle is made of value, quality, and also adapting the traditional triple constraints; it is represented in Figure 10. Value is considered as the most important fixed element, and it should be created incrementally rather than only at the final delivery of the product. Creating

value incrementally makes the value to be present from the start. The traditional project constraints are still important, but the aim is to deliver the value and quality within the traditional constraints of scope, time, and cost.

Figure 10 Transformation of traditional constraints into agile (Highsmith (2009) in Hendriks, 2019)

When compared with the traditional way of working, the agile way of working is very different. Table 3.2 shows some fundamental differences between both approaches based on the study of Hoda et al., (2008).

Table 3.2 Difference in concept between traditional and agile approach (Hoda et al., 2008)

Categories	**Traditional approach**	**Agile approach**
Development model	Linear/ sequential	Iterative and adaptive
Focus	Process	People
Management	Controlling	Facilitating
Customer involvement	Requirements gathering and delivery phases	constantly involved
Developers	Work individually within the teams	Collaborative
Technology	Any	Mostly object-oriented
Product features	All included	Most important first
Testing	End of the development cycle	Iterative
Documentation	Thorough	Only when needed

Agile methods do not give much importance to the documentation of the process, unlike traditional management (Gary et al., 2011). The Agile method has smaller phases compared to the traditional approach (Lima et al., 2015). Also, when using an agile process, the team tends to make quicker decisions (Cohn & Ford, 2003).

Adding to the differences in the agile approach when compared to the traditional approach, the study of Fuchs & Golehhofen (2019), mentions the following for an agile approach:

- Planning is based on the latest updates
- Delivery of values is time-boxed and is based on the customer's priorities
- Focus is on continuous interaction and satisfaction of the customer
- Responds to changes in the project based on adaptive planning and action
- A collaborative approach by the self-organized team
- Innovative, experimenting and it learns from the failures

With all the differences mentioned above from various literature, it is evident that iterative development is better and more effective than the traditional project management approach. It also enables continuous and fast delivery compared to the delivery of products by the traditional approach. Figure 11 illustrates the difference in the product development by traditional and agile project management approach based on the study from Fuchs & Golehhofen, (2019).

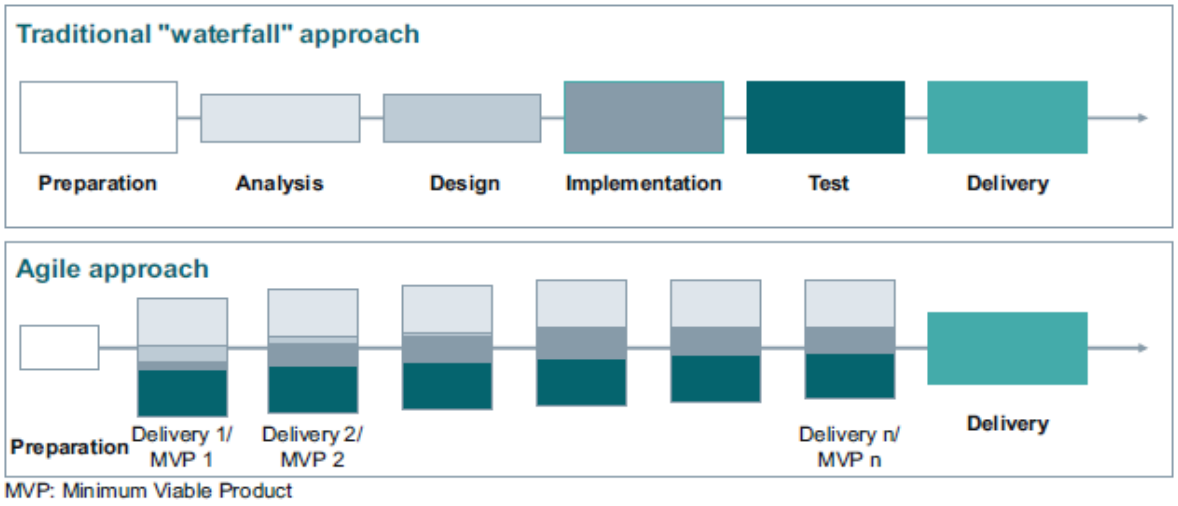


Figure 11 Difference in product development between traditional and agile approach (Fuchs & Golehhofen, 2019)

According to the values and principles of agile, various frameworks are developed. To name a few: Scrum, Feature-Driven Development (FDD), crystal, Dynamic System Development Method (DSDM), eXtreme Programming (XP) (Hoda et al., 2008). The study from Ebert & Paasivaara, (2017) introduces new frameworks like Scrum of Scrums (SoS), Scaled Agile Framework (SAFe), Large-Scale Scrum (LeSS), Disciplined Agile Delivery (DAD), Lean Scalable Agility for Engineering (LeanSAFE). Among all the mentioned frameworks of agile, the focus of this research is on the Scrum framework as the representatives of the company were interested in the same.

3.4 Scrum method

The authors Schwaber & Sutherland acquired the findings of Takeuchi and Nonaka, who first coined the term Scrum in 1986 and developed it in the 1990s. According to Schwaber & Sutherland (2013) in Weinreich *et al.* (2015), Scrum is defined as "*a framework within which people can address complex adaptive problems while productively and creatively delivering products of the highest possible value*." It is a framework for product development made up of various processes, and techniques, which also applies to the complex products (Streule et al., 2016). Scrum covers mainly the attributes related to project management (Dingsøyr et al., 2008). It is iterative, commonly used, and simpler than other frameworks of the agile project management approach (Eloranta et al., 2016). The key feature of Scrum is flexibility, self-organization, and autonomy (Hidalgo, 2019). The main aim of Scrum is to make the product development process more efficient and reduce the time to market (Cervone, 2011). It incrementally delivers value while adapting to complex problems.

Scrum is commonly used for software development due to its evidence of increased productivity and efficiency. It also increases the quality of work by the development teams. Integrating Scrum in the development process allows innovation in the process and aids in handling the uncertainty (Böhmer

et al., 2017). The main challenge is to estimate the complexity of the task (Garzaniti et al., 2019). According to Cho (2008), Scrum employs the following three underlying concepts in its implementation, which are transparency, inspection, and adaptation, and are achieved by the iterative process, roles, and elements of Scrum (Weinreich *et al.*, 2015).

- *Transparency* - All the aspects of the process that affect the outcomes are visible to all.
- *Inspection* - Frequent inspection of various aspects of the process is done to avoid/quickly remove the unacceptable variance.
- *Adaptation* - If some aspects of the process are unacceptable, then the process needs adjustment.

3.4.1 Scrum roles

There are three roles in the Scrum framework, i.e., the Scrum Master, the Product Owner, and the scrum team, and each role has various responsibilities (Eloranta et al., 2016, Cho, 2008).

Scrum team- It is a self-organized and coordinated team with members of different backgrounds and skills. The team is responsible for selecting their tasks for each sprint and managing to complete it within each sprint (Deemer et al., 2012). Their primary duties are:

- To fulfil the requirements of the product owner
- To complete all sprint items during sprints

Scrum Master **-** The Scrum Master is the coach of the team and also the moderator between the team and product owner (Weinreich *et al.*, 2015). Anyone from the team members can become a Scrum master but not the product owner. For bigger teams, it is advisable to have a dedicated member for this role (Deemer et al., 2012). The primary duties of a Scrum master are:

- To facilitate the daily stand up and sprint reviews/retrospectives
- To eliminate any hindrance and be responsible for carrying out the Scrum process smoothly
- To ensure the team follows the values and principles of agile

Product owner - The Product Owner is the voice of the client. Also, acts as the common link between the team and stakeholders (Deemer et al., 2012). The product owner evaluates the outcome of each sprint and decides on the changes in the product. The primary duties of a product owner are:

- To create and manage the product backlog
- To prioritize the requirements of the product that should be achieved in the sprint
- To set targets for the upcoming sprints

3.4.2 Scrum ceremonies

There are four types of meetings that take place in the Scrum, and it is the Scrum master's responsibility to chair all the meetings (Schwaber & Sutherland, 2017).

Sprint planning meeting is to plan the sprint, usually lasts for some time depending on the size of the team and the length of the sprint, and conducted at the start of the sprint. According to Schwaber & Sutherland (2017), the meeting divides into two segments, in the first segment, it is mainly the

discussion between the product owner and the team about the overall goal and the objective of the particular sprint, which has to be achieved. The team needs to understand and take account of possible limitations. In the second segment, it is when the team picks items from the product backlog to the sprint backlog. Everyone picks the maximum amount of tasks that can be completed by them during the sprint.

Daily stand up meeting is a short meeting with the Scrum team except for the product owner for approximately fifteen minutes and conducted daily. The main aim of this meeting is to update every member of the team about the progress in the sprint and also to share and learn information from each other (Schwaber & Sutherland, 2017). It is to plan the day for each member of the team and discuss the happenings of the previous day.

Sprint review meeting is conducted at the end of every sprint with the whole Scrum team but can also include the customers and other stakeholders. The main aim is to review the work done at the end of the sprint and further discuss with the product owner about the necessary changes or additions needed to the product backlog (Schwaber & Sutherland, 2017).

Sprint retrospective meeting is planned after the sprint review meeting, and it is for the Scrum master and Scrum team. The main purpose of this meeting is to analyze the whole process of sprints, reflect on both issues and positive outcomes, and learn for the future (Schwaber & Sutherland, 2017). It is to evaluate all the members, processes, and the interactions between them critically (Streule et al., 2016).

3.4.3 Scrum process

The process of Scrum, according to Eloranta et al. (2016), begins when the product owner creates a product backlog (to-do list) for the project based on the needs of the client. The product backlog consists of all the requirements suggested for the implementation of the software. After that, the product backlog is refined in a meeting with the Scrum team. The discussion in the meeting is about the splitting of complex tasks into sub-tasks or about adding more tasks to the list. All the items/tasks are the functionalities of the product developed. The items in the Backlog are stored as a short written description of the requirements (who, what & why) and are called user stories are implemented within a period called a sprint (Heikkila et al., 2013). A sprint usually lasts for a few weeks. Every sprint is one project from the scrum point of view (Maximini et al., 2018), which means a sprint requires finishing of all sprint tasks within the prescribed time of the sprint. For every sprint, few of the items are chosen from the product backlog, which is called the sprint backlog. The sprint backlog is the collection of items that are completed in the current sprint. The items in the backlog split into various tasks with the description and requirement for the item to be marked as completed (Schwaber & Sutherland, 2017). For sprint planning, the items (tasks) are taken from the product backlog (Garzaniti et al., 2019). The selected tasks are the tasks that the Scrum team considers that can reach the state of done in the sprint. The state of done is when the items from the sprint backlog are considered to be finished then the item moves from the sprint backlog (Streule et al., 2016). The sum of all the items (tasks) that the Scrum team considers done form the increment.

The study of Streule et al. (2016) suggests the use of planning poker, an activity used to estimate the number of items selected from the product backlog to work on in the sprint, and the members of the Scrum team decides it. According to Loffler et al. (2010), many testing activities are conducted for the

software during the sprint: (1) Unit testing; a test to check the correctness of the new item, (2) Regression testing; a retest to check the correctness of the old items after changes, (3) Integration testing, a test to check the integration of old and new items in the system, (4) Acceptance testing; a test to check whole software based on the requirements of the user. To finish the sprint, the testing has to be completed and assure that the software is integrated well. There are many automated tools available to conduct the testing activity efficiently.

All the members of the team update the sprint backlog daily by reporting the amount of time taken by them in doing the task. Further, the estimates are made based on the time that is needed to finish the task. The update by all the team members is visualized on a graph called sprint break down chart where the amount of work that is left to complete all the tasks in the current sprint for the whole team can be seen (Deemer et al., 2012). The overview and progress of the sprint are visualized through the sprint burndown chart, and if something goes wrong, then necessary changes are made to the plan accordingly. Each sprint aims at producing a complete deliverable, and all the deliverables together form the product by increment (Schwaber & Sutherland, 2017). There is no limit to the number of items in the product backlog; the only condition is that no element that cannot finish in one sprint be included. The rate/velocity of the development in the sprint is measured by the story points per sprint and is called the development velocity (Heikkila et al., 2013). Also, the study mentions release planning, which is a practice where the decision of what features to be released next is taken. It is about planning on how to increase efficiency with long term planning. Figure 12 represents the process in the Scrum.

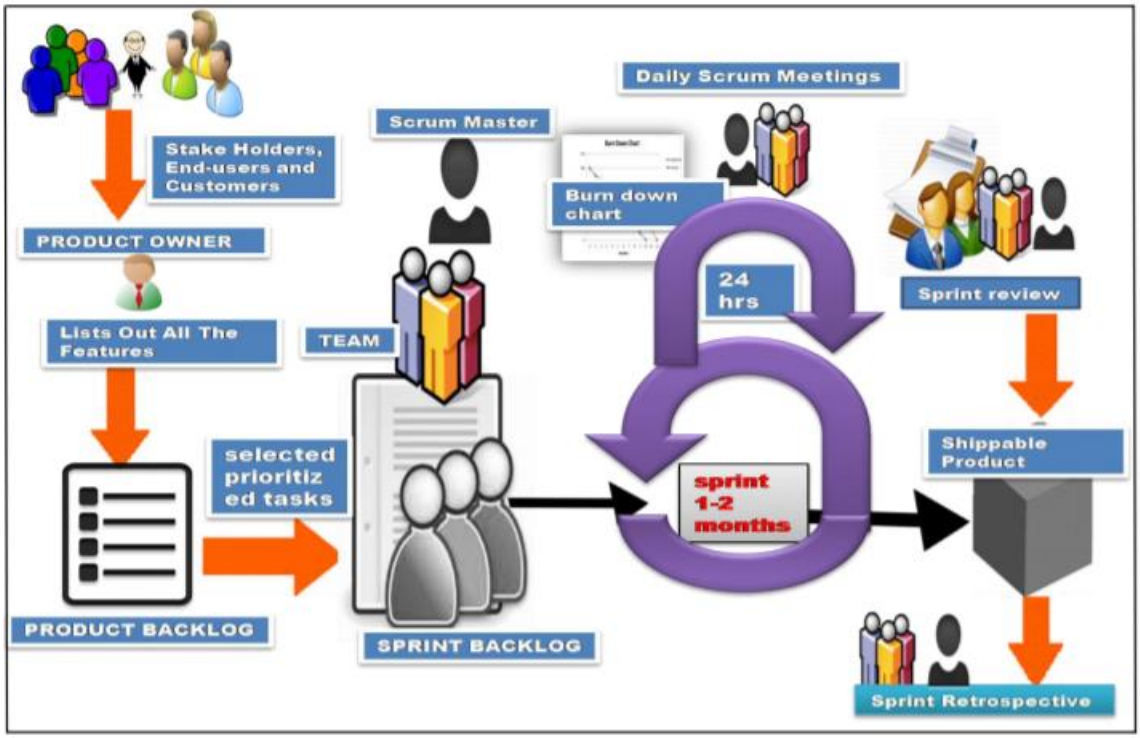


Figure 12 Scrum method (Deemer et al. (2012) in Morampudi et al., 2013)

3.4.4 Benefits of Scrum

Scrum is valuable when used for complex products, especially in those cases when there is a new technology or growing requirements involved in a project (Streule et al., 2016, Mushtaq & Qureshi,

2012). Overall, Scrum is effective and flexible for both small, as well as large projects. The only difference is that small projects can use the techniques of Scrum directly, whereas large projects should split into small projects and then use Scrum (Livermore, 2007). It has the potential to deliver high value in the following cases, (1) when the team members have less experience in working together, (2) when increasing complexity affects the cost of the project, and (3) when there is a lack of understanding of the problem and solution (Tomar, 2017). Various works of literature are summarised to understand the overall benefits of Scrum and are summarised in Table 3.3

Table 3.3 Benefits of Scrum (Permana, 2015, Maximini et al., 2018, Cooper & Sommer, 2016, Mahnic, 2011, Mushtaq & Qureshi 2012, Bianchi et al., 2020, Ciric et al., 2018, Schuh et al., 2016)

Benefit of Scrum	Source
• Ability to manage the changing requirements • Improved visibility in project • Flexibility in the development process • Improved communication between the team members • Early and steady feedbacks of customer integrated into the development process	Maximini et al. (2018)
• Detailed estimations for tasks • Improved quality of the deliverable	Permana (2015)
• Reduced the amount of documentation	Cooper & Sommer (2016)
• Improved planning and estimation • Reduces time and resources	Mahnic (2011)
• Focus on integration/interfaces in the project • Evaluated work throughout the process	Mushtaq & Qureshi (2012)
• Controls scope creep	Bianchi et al. (2020)
• Reduces cost and time taken in the project • Reduces the time required to plan • Gives the flexibility to manage uncertainty due to innovation • Higher effectiveness to the predevelopment stages • Achieves customers expectation • Reveals deficiency early in the process • Creates project plan with shared responsibility	Ciric et al. (2018)
• Less detailed planning, specification, and documentation • Early testing of physical functional prototypes • Member of the project has a strong centralized work environment	Schuh et al. (2016)

3.4.5 Scrum practices

The thesis work of Hendriks (2019) summarises various agile practices with the focus on the Scrum method, and then all practices (elements) of the Scrum were categorized to the respective agile values and principles. Table 3.4 consists of the synthesized commonly applied list of Scrum elements/practices based on literature study done Hendriks (2019).

Table 3.4 Scrum elements related to agile values and principles (adopted from Hendriks, 2019)

Agile values	Agile principles	Scrum elements
Individual and interaction	Business people and developers work together	Scrum of Scrums
	Motivated individuals	A shared belief in agile within the team
		Daily stand-up/Scrum meetings
		Coaching of agile
	Face-to-face communication	Informal face to face communication
	Self-organizing team	Multidisciplinary team with one goal and Division of roles
	Reflection	Sprint reviews/retrospectives
Working software	Working software	Acceptance tests
	Technical excellence and enhanced agility	One defined process/ Scrum way of working
Customer collaboration	Satisfy customer	Frequent communication with the customer
Responding to change	Welcome to changing requirements	Sprint planning and selection of work
	Frequent delivery	Prioritised Backlog
		Time boxed sprints
	Simplicity	Informal design
	Sustainable development	Virtual Scrum board
		Burndown chart
		Release planning

Despite the differences in HW development compared to SW development, more and more companies from the hardware sector realize the value of agile (Hanser, 2010). According to Garzaniti et al. (2019), the agile method is efficient, suitable for hardware development and defined to achieve the deliverables by main milestones. Therefore, the next step for discussion is to find out which practices of Scrum mentioned in the above Table 3.4 can be applied for HW development projects.

3.5 Scrum for HW development

According to Gloger & Häusling (2011), though the Scrum method is standardly used for software development, it can also be used for hardware development. To get the full potential of the Scrum method in hardware development, some adjustments in the practices are needed (Augustin & Schabacker, 2019). Therefore, to get complete advantage of the method, it is important to tailor the parameters to the need/type of the project.

According to the study of Lima et al. (2015), for products with HW and SW integrated, the Scrum method can be adapted by adjusting according to the needs of the project. Using the Scrum method in HW development makes each phase smaller and delivers value. The study also shows that quick response to changes can be achieved by reducing the development cycles and integrating both hardware and software teams. The study also mentions that the unit test is a good practice, and when applied to the hardware development will help in giving a quick response to the errors and

reduce the failures. Upfront prototyping is something to learn when compared to the traditional approach where the prototyping is at the final validation (Punkka, 2012).

From section 3.1.1, we could understand the nature of HW development, so the next question is how scrum practices could be applied to HW development considering the differences with SW development. So to understand that it is important how the agile values and principles can be transformed into HW development.

3.5.1 Agile values and principles for HW development

The study of Fuchs & Golehhofen (2019), interprets how the agile values could be applied to the HW development and is represented in Table 3.5.

Table 3.5: Agile values for HW development (Fuchs & Golehhofen, 2019)

Agile values	Application on HW development
Individual and interaction	This value applies as it is to HW, but since a lot of different domains are involved in the HW development projects, this leads to an increased number of interactions and overall project complexity. Therefore it is necessary to have a framework to manage and reduce complexities.
Working software	Creating a continuous working product in HW is not as easy as it is in SW. Compared to SW development, documents in HW development projects are more critical and mandatory. Also, knowledge management is an essential aspect of the documentation of HW; therefore, it is necessary to store all the information for a long time in a proper way.
Customer collaboration	This value applies as it is to HW. Constant interaction and early feedback from the customer is the key to satisfy them. This will help in forming the right goal and objectives and also reducing any politics in the company.
Responding to change	The differences in HW development when compared to SW development clearly say that it is difficult to respond to changes in HW because of higher complexity and high involvement of material cost. Therefore a clear strategy or a methodology is required for when the changes can be introduced in the development process.

Some agile principles can be applied directly to HW development, but some principles require to be adapted when applied to HW development, and this is represented in Figure 13 based on the study of Fuchs & Golehhofen, (2019). Moreover, the study also discusses changes that should be taken in HW development for the agile principles that cannot be applied to HW development directly and are explained in Table 3.6.

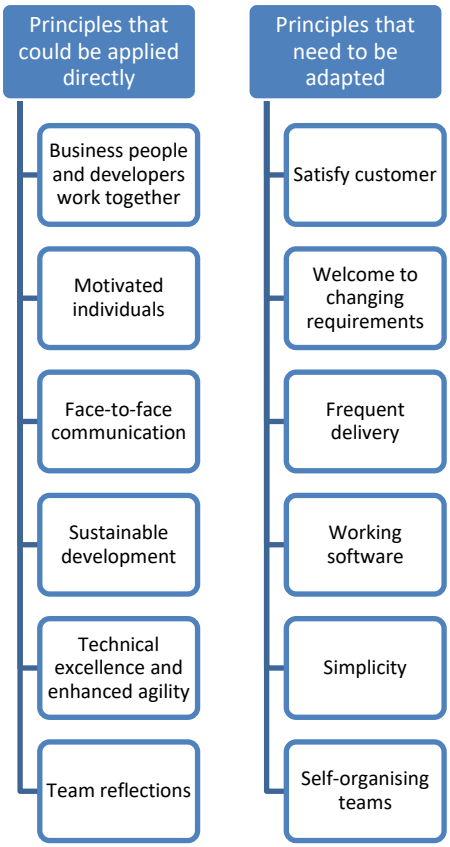


Figure 13 Agile principles based on HW development (own illustration based on Fuchs & Golehhofen, 2019)

Table 3.6 Agile principles for HW (Fuchs & Golehhofen, 2019)

Agile principles	Application on HW development
Satisfy customer	This principle could be applied in HW if the customer for whom the working software is delivered is considered as an important stakeholder for who benefits from the outcome rather than just an end customer. It is critical to delivering working output, and therefore it is necessary to take inputs from the customers early in the development process to reap the benefits.
Welcome to changing requirements	The goal for modern HW development is to allow late changes in requirements, and this is not possible when requirement specifications are fixed in the early phases. Therefore it should be determined which requirements are stable and which are more prone to changes to increase the flexibility towards the needs of customers.
Frequent delivery of working output	Creating a working output/product in HW is difficult compared to SW. Therefore, to achieve this principle, the output should be considered as something that could be discussed at the end of the sprint instead of working deliverable. Also, the duration of the development cycle should depend on the expected outcome and decided at the start of the cycle.
Working software	It should be considered as an output that could be discussed, reviewed and improved. The growth of maturity in the output will mark as the progress for the output.

Simplicity	It means to focus on what matters and adds value to the output. Therefore the importance is to be given to the customer requirements.
Self-organizing teams	This principle implies that creating the right team that is empowered to innovate is what is needed. A correct set of members of the team will be able to create new ideas and approaches to solve a problem. The team should combine have what it takes to solve a problem as this will help fix the core cause of the problem.

Based on the discussion in Table 3.6, it is clear that the agile principle; "working software" and "simplicity" suits specifically for SW development and therefore needs to change for HW development. The scrum practices related to "working software" and "simplicity" are acceptance tests and informal design, respectively (refer to Table 3.4). Therefore, those scrum practices are not considered for their application in HW development. Also, though the agile principle "Business people and developers work together" can be applied directly to HW development, but the scrum practice related to that agile principle; "scrum of scrum" is not considered for its application in HW development, as it is a high-level practice of scrum.

3.5.2 Scrum practices for HW development

The Scrum elements are the artefacts of the Scrum framework. These practices are discussed next.

Face-to-face communication is an effective way to convey information within the Scrum teams as it is a more powerful way of communication than written documents. Also, taking continuous feedback from the customers makes the project efficient and delivers a better quality product in the end (Cooper & Sommer 2016). However, for the success of Scrum, the active involvement of the customer in the development process is considered very crucial (Hanslo & Mnkandla, 2018). The verification and validation phase is the key stage in hardware development; however, testing can be avoided by incremental development and continuous feedback (Garzaniti et al., 2019).

The study of Passivara et al. (2008) focuses on the condition when face-to-face communication is not feasible due to the involvement of different locations and different time zones, in that case, it is difficult to conduct the planning meetings as it is difficult for everyone to participate. However, the involvement of all team members is crucial to get everyone's opinions. Therefore, Passivara et al. (2008) suggest having a pre-work done aiming to solve most issues of the customers and developers before the actual planning starts and make the actual meeting short and focused. The study of Passivara et al., (2008) points out various challenges of distributed team members relating to communication and team management and suggests agile practices as a solution to develop better relationships and optimize cultural exchange.

According to Marcal et al. (2007), the team meets during daily stand-up to communicate and to track the progress of work. With daily stand up, the team also could resolve issues. From the study of Hidalgo (2019), it is clear that the stand up mainly focuses on tracking progress and communicating with others. The stand-up also helps to resolve any emerging issues or hindrances in the process. The study of Passivara et al. (2008), shows that daily stand up can also help in solving language barriers in the team. The daily meetings of Scrum can significantly reduce the duplication of work caused by a lack of communication between team members (Cho, 2008).

Further to document all the elements, the Scrum masters use the Scrum boards, which help to understand the social aspects of the tasks clearly (Sharp et al., 2009, as cited in Hidalgo, 2019). According to Anderson et al. (2012), as cited in Hidalgo (2019), the board acts as a tool for tracking all the activities implemented daily and is considered very useful because of its practicality to visualize the deliverables. The digital/virtual Scrum boards are similar to physical Scrum boards. According to Garzaniti et al. (2019), to track tasks, and the status of development, Jira (virtual Scrum board) is used. Jira graphically represents the development and shows the distribution of the tasks in the team. It makes the workflow in the project visible to all the members and shows tasks of all the team members, which aids them in visualizing the priorities. With the boards used virtually or physically, it is easier to visualize the bottlenecks and optimize the efforts (Coco et al., 2011 as cited in Hidalgo, 2019).

Scrum meetings are a way to improve communication and coordination within project teams. It is not just useful within teams but also for the management of cross teams (Passivara et al., 2008). According to Mushtaq & Qureshi (2012), the advantage of conducting Scrum meetings is that it helps to focus on the integration and overlapping areas and to know the project status. Also, it mentions that with the review meetings, the team examines the work done/implemented during the sprint.

Sprints improve the ability of planning and estimating, especially for beginners (Mahnic, 2011). When using high-frequency sprints with reduced scope, it reduces the time and resources applied while it also increases the flexibility to adjust to changes in design with the emerging context of the project (Bianchi et al., 2020). The study also shows that reduced scope in sprints leads to a more accurate estimation of resources needed in meeting requirements.

Timeboxing of sprints facilitates monitoring of the development process and avoids scope creep by giving the highest priority to achieve functionalities (Bianchi et al., 2020). It also helps in making a real commitment and achievable plan. Timeboxing aids in keeping time targets, and in projects, it is advisable to keep a target set for time, cost, quality, and feature required. For HW development, time constraints of sprints may not work all the time, especially when the waiting time for physical prototypes is because of external dependencies. Therefore, the sprints need to be flexible to tackle the delays (Augustin & Schabacker, 2019). Only selected techniques of the Scrum method will be suitable due to the difficulty of delivering working products incrementally. According to Garzaniti et al. (2019), there will be a difference in time management when applying the Scrum method for HW development compared to SW development. In SW development, the planning is dependent on the team, sprint structure, and the Scrum master, and they have control of the development. Whereas in hardware development, there are external dependencies involved, and the team is not in direct control of it. In that case, the strategy is to schedule the overall workflow considering the lead-time and different scenarios.

A prioritized list of features for the product, consisting of product requirements evolves with the sprints (Cooper & Sommer, 2016). According to this study, using backlogs gives a visual plan of the development with an ability to adapt to changes in the process of development. To monitor constantly, to track the development, and to assess the correct implementation, the team can use various tracking technologies. Project planning in Scrum uses lightweight techniques like burndown charts (Sutherland, 2001). The scope fulfilment throughout the project can be realized by following the burndown chart (Mahnic & Zabkar, 2012).

According to Maximini et al. (2018), Scrum helps in identifying any weakness or inadequacy in an organization but does not solve the problem. He further suggests that a lack of knowledge about Scrum will lead to unclear roles and responsibilities when implementing Scrum. Therefore, it is important to have full knowledge of Scrum and coaching before implementing it. He also mentions that member working on multiple projects at the same time affect the productivity (loss of 20% for each new project) of the team (Weinberd 1991 as cited in Maximini et al., 2018). The following are the problems due to a lack of knowledge of Scrum based on Maximini et al., (2018).

- Non-existence of product owner or Scrum master does the role of product owner, in both cases, it will lead to conflicts in roles and inadequate preparation for the requirement of the product.
- Non-existence of a Scrum master/ developer replaced as Scrum master will affect the success of the project, especially for the pilot projects as the role of a Scrum Master is very crucial for proper implementation of the Scrum process.
- Lack of development skills in the team is a significant problem for working together as a team towards one goal. Therefore, it is important to identify all the important skills required before the start of the project. The hiring of an external consultant will solve the issue for a short time but not the problem.

According to Ovensen & Dowlen (2012), time consumed in building the prototype of hardware is higher than the time consumed to write or compile software; therefore, frequent delivery in an incremental manner during the short iteration is still one of the major challenges. Also, it is difficult and challenging to break the product in small tasks for incremental delivery, as HW lacks modularity. Therefore applying agile methods for HW development also has challenges, and it is specific to four main domains: *Constraints in physicality, Mindset, Team distribution, and Scaling* due to the differences in HW development compared to SW development (Atzberger & Paetzold, 2019).

3.6 Theoretical framework

Based on the literature review on the applicability of Scrum for HW development, it is seen that not all practices mentioned in Table 3.4 apply, and some of the practices need customization for HW development. Therefore, a theoretical framework is made based on the benefits of the Scrum practices that have applicability in HW development (refer section 3.4 & 3.5), summarised in the upcoming Table 3.7. In Table 3.7, for each scrum practices, whether it can be applied directly for HW development or it should be adapted is decided based on the agile values and principles, it is related to (refer to section 3.5.1).

- *A shared belief in agile within the team & Coaching of agile*

  It helps in identifying any weakness or inadequacy in an organization and gives clarity on various roles and responsibilities (Maximini et al., 2018).

- *Daily stand-up/ scrum meetings*

  The team meets during daily stand-up to communicate and to track the progress of work. With daily stand up, the team also could resolve issues (Marcal et al., 2007). It mainly focuses on tracking progress and communicating with others. The stand-up also helps to resolve any emerging issues or hindrances in the process (Hidalgo, 2019). Daily stand up can also help in solving language barriers in the team (Passivara et al., 2008). It can significantly reduce the

duplication of work caused due to lack of communication between team members (Cho, 2008). It is not just useful within teams but also for cross-team management (Passivara et al., 2008). According to Mushtaq & Qureshi (2012), the advantage of conducting scrum meetings is that it helps to focus on the integration and overlapping areas and to know the project status.

- *Informal face to face communication*

Face-to-face communication is an effective way to convey information within the scrum teams as it is a more powerful way of communication than written documents (Schuh et al., 2016).

- *Multidisciplinary team with one goal and Division of roles*

It increases motivation and a sense of ownership within members (Eloranta et al., 2016). It also improves involvement in the team, which enhances knowledge sharing (Cho, 2008).

- *Sprint reviews/retrospectives*

With the review meetings, the team examines the work done/implemented during the sprint (Mushtaq & Qureshi, 2012). According to Overhage al. (2011), retrospective and review are similar to the lesson learned meetings and facilitates the learning process.

- *Frequent communication with the customer*

Continuous and early feedback from the customers make the project efficient and delivers a better product in the end (Cooper & Sommer 2016). It reveals deficiency early in the process (Ciric et al., 2018) and enhances the quality in the deliverable at the end (Permana, 2015).

- *One defined process/ scrum way of working*

Following a defined scrum way of working for HW development makes each phase smaller and delivers value (Lima et al. 2015). The study shows that quick response to changes can be achieved by reducing the development cycles in HW development.

- *Sprint planning and selection of work*

Sprints improve the ability of planning and estimating (Mahnic, 2011). It reduces the time and resources applied, while it also increases the flexibility to adjust to changes in design with the emerging context of the project (Bianchi et al., 2020). It leads to an accurate estimation of resources needed in meeting requirements (Bianchi et al., 2020).

- *Prioritised Backlog*

It gives a visual plan of the development with an ability to adapt to changes in the process of development (Cooper & Sommer, 2016). It is refined for every sprint and maintains clear requirements (Garzaniti et al., 2019). It also gives the ability to adapt to changes in new sprints (Maximini et al., 2018).

- *Time boxed sprints*

It monitors the development process, avoids scope creep, and helps in making a real commitment and achievable plan (Bianchi et al., 2020). It aids in keeping time targets in projects (Ciric et al., 2018).

- *Scrum board*

It helps to understand the social aspects of the tasks (Sharp et al., 2009, as cited in Hidalgo, 2019). It acts as a tool for tracking all the activities implemented daily and considered very useful

because of its practicality, visualizes the deliverables (Anderson et al., (2012) as cited in Hidalgo, 2019). It is used to track tasks and the status of development. It makes the workflow in the project visible to all the members and shows tasks of all the team members, which aids them in visualizing the priorities (Garzaniti et al., 2019). It also makes it easier to visualize the bottlenecks and optimize the efforts (Coco et al., 2011 as cited in Hidalgo, 2019).

- *Scrum tools (Burndown chart/ Release planning)*

  It is used to monitor constantly, track the development, and assesses the correct implementation (Sutherland, 2001). The scope fulfilment throughout the project can be realized by following the burndown chart (Mahnic & Zabkar, 2012). Further, the updates on the estimates are visualized, which helps to track the time and progress of tasks in the current sprint for the whole team (Deemer et al., 2012).

Further, the applicability of scrum practices for HW development is decided based on the agile value and principle; each Scrum practice is related to (refer section 3.5.1). The benefits known from the theoretical framework will be used as background knowledge to investigate the fit of Scrum elements in the current management approach of the company. The theoretical framework will be used as a guide for the case study.

For this purpose, case studies will be analyzed on different themes in the traditional project management approach. The themes are chosen based on specific aspects of the project, where the benefit of Scrum is expected based on the advantage of the scrum method studied. The chosen themes are project planning, project scope, time, cost, quality standards, task definition, requirements management, deliverables, project evaluations, and lastly, engagement with customer and team (refer appendix B). Further, since the interface between HW & SW development plays a major role in the project; therefore, it is also added as one of the themes. Each theme will be investigated based on how it was done at the time of the project. The themes are analyzed by questioning the interviewees about the management approach for each theme about how it took place and what methods or strategies were applied.

For the success of HW development projects, the management approach plays an important role. The selection of the right management approach is dependent on the project characteristics and complexities. Therefore, it is important to understand the complexities present in the HW development projects. For this purpose, a complexity framework (TOE) for the process industry by Bosch-Rekveldt (2011) is used to assess the complexities of the project. It consists of 47 elements of complexity and is divided into three subcategories of complexity called Technical (T), Organizational (O), and External (E). It helps in establishing the complexity footprint of the project by questioning the respondents to scale all the elements on a 5-point Likert scale ranging from zero to four based on the perceived contribution of each element in the project. Finally, the suggestions on the applicability of Scrum elements for HW development projects will be drawn based on the finding of the cases.

Table 3.7 Theoretical framework

Agile Values	Agile Principles	Scrum elements for HW development	Benefits	For HW development –can be applied /should be adapted
Individuals and interaction	Motivated individuals	A shared belief in agile within the team	• Clear roles and responsibilities • Helps in identifying any weakness or inadequacy in an organization	Applied directly
		Coaching of agile		
		Daily stand-up/ Scrum meetings	• Track progress of work • Resolve issues and hindrance in the process • Resolve language barriers • Reduce duplication of work • Improves communication &coordination within teams • Improves the correlation /cross-team management • More focus on integration and overlapping areas. • Project status is known to all the team members	Applied directly
	Face to face communication	Informal face to face communication	• Effective mode to convey information	Applied directly
	Self-organizing teams	Multidisciplinary team with one goal and Division of roles	• Improves team involvement • Enhances knowledge sharing • Increases motivation and sense of ownership	Adapted
	Reflections	Sprint reviews/retrospectives	• Improved learning • Examine the work done	Applied directly
Customer collaboration	Satisfy customer	Frequent communication with the customer	• Early feedback from the customer • Aid for better quality in the deliverable	Adapted
Working software	Technical excellence	One defined process/ Scrum way of working	• Smaller phases and delivers more value • Quick response to changes • Reduced development cycle	Adapted

Agile Values	Agile Principles	Scrum elements for HW development	Benefits	For HW development –can be applied /should be adapted
Responding to change	Welcome to changing requirements	Sprint planning and selection of work	• Improves ability to plan & estimate • Reduce time and resources applied on a task • Improves the flexibility to adapt to changes in the project	Adapted
	Frequent delivery	Prioritised Backlog	• Visualize the plan for development • Ability to adapt to changes in the development process • Clear requirements	Adapted
		Time boxed sprints	• Monitor the development process • Avoids scope creep • Realistic & achievable plan • Enhances the time target	Adapted
		Scrum board	• Understand the social aspects of tasks • Aids in progress tracking/monitoring of tasks • Visualise deliverables • Overview of the status of development/tasks • Shows the clear distribution of tasks • Visible workflow &priorities	Adapted
		Scrum tools (Burndown chart/ Release planning)	• Tracking of the development • Assess right implementation in the development • Track scope fulfilment • Track time and progress of tasks	Adapted

4 Case study

The goal of this chapter is to explore and describe the various projects of the company and understand how they are managed. The chapter is constructed in the following manner; section 4.1 explains the case study outline. It contains criteria for case selection followed by the description of the selected cases and the approach for the analysis. Section 4.2 describes project A, case analysis, and finally followed by finding issues of the case. Similarly, section 4.3 and 4.4 are dedicated to the discussion, analysis, and formulating issues of project B and project C, respectively.

4.1 Case study outline

The motive of this section is to explain the method for the selection of cases based on the case selection criteria. Section 4.1.1 is for the formulation of case selection criteria based on the initial discussions at the company. Section 4.1.2 is for explaining the case selection process based on the criteria and describing the selected cases. Finally, Section 4.1.3 contains the approach for the data collection from the selected cases.

4.1.1 Case selection criteria

At the start of the research, the researcher held discussion meetings with few important project leaders from Build-to-Specification and Installed base projects. The motive of the meeting is to get an initial understanding of the projects in the company. Based on the discussions, the researcher formulates the following criteria for the selection of cases.

- **Recently finished projects**

The selected project needs to be recently completed because fetching information about the project that is completed long back is difficult to get. It is expected that the employees tend to be more objective when they are not closely involved with the project anymore. Also, studying recent projects gives a clear overview of the current scenario.

- **Availability of key members and project documents in the company**

It is relevant that people/information are still available at the company so that the researcher can obtain important information related to the case.

- **Critical to manage project**

From the initial discussion at the company, it is realized that few complexities in the, like the involvement of volatile requirements, first of a kind project, international clients/culture, and fixed budget, make it difficult to manage such projects. Therefore, it would be interesting to study projects, which involve any of these complexities and know how the company manages such projects.

4.1.2 Case selection

To select the cases for the research, suggestions from the major project leaders at the company were taken. A case selection meeting was held at the company with the top four project leaders. The representative of the company suggested the four top project leaders for the discussion. In the case

selection meeting with the project leaders, a presentation was given by the researcher explaining them the following points:

- why a case study is needed
- what are the selection criteria
- what is the aim of the case study
- What are the case selection criteria

After the presentation, the projects from the company were evaluated based on case selection criteria. Three cases were selected; two from Build-to-Specification and one from Installed-base projects. Table 4.1 represents the list of the selected projects and a brief description of the management approach applied in the project given by the project leaders at the case selection meeting.

Table 4.1 Description of selected cases

Case	Project name	Type of the project	Description
1	A	Build-to-Specification	HW development is done completely by a Traditional (waterfall) approach, and SW development is done completely with the Scrum method
2	B	Build-to-Specification	HW development is done in a dynamic approach (partly with Scrum and partly with a waterfall approach) and SW development is done completely with the Scrum method
3	C	Installed-base project	HW development is done by the traditional approach, and SW development is done completely with the Scrum method.

4.1.3 Data collection

All the data for each selected case is from the available general documents related to the project, complexity assessment by TOE framework (Bosch-Rekveldt et al., 2011), and by conducting interviews with the members from the selected project. The project documents are collected from the project leader of each case. From each case, three members were selected to do the complexity assessment and for the interviews. Three members include the project leader, one key member from the HW development team, and one key member from the SW development team. The researcher selected the other two key members based on the suggestion of the project leader of each case. The complexity assessment is conducted before the interview because the answers to the interviews should not influence their assessment of complexity. Therefore, the researcher sent the TOE framework for complexity assessment to each selected interviewee a week before their interview to do the complexity assessment. The interview has an explorative nature to extract as much information as possible from all the interviewees on the management approach used.

From each case; three members, including the project leader, so in total, nine interviews are conducted. For the case study, a semi-structured interview was chosen, lasting for approximately one hour to get maximum details about the management approach of each project. A mail was sent to all interviewees, explaining the purpose of the interview and the name of the project for which they

were selected. Further, an appointment was made with all the interviewees. The questions for the interview were divided into three parts (A, B & C). The part A of the interview questions was focused on extracting information about the interviewees and basic details about HW and SW development of the project. The part B of the interview questions was framed based on the themes in the project management approach where the benefit of Scrum is observed and also about the interface where HW and SW coexists. Part C was about asking the interviewees about how the complexity elements that were marked high (three or four) in the complexity assessment were managed. All the interview questions can be found in Appendix C. All interviews are recorded, transcribed by the researcher, and later validated by the interviewees. All the interviews were face to face interviews conducted in the company. The selected cases are further analyzed in the upcoming sections 4.2, 4.3, & 4.4, respectively. Later cross-case analysis is performed in chapter 5.

4.1.4 Intra case analysis

All the cases are analyzed based on a general description, the project complexity, and the project management approach. To analyze each case, the researcher used the analysis protocol, and it is described as the following:

General description

A general description of the project based on the documents gathered: the project management plan, the statement of work, and presentations made by the company (if available). Also, the answers of part A of the interview questions are analyzed to add to the case description.

Project complexity

From the results of the TOE framework filled by three interviewees of each case, the complexity element that is marked high (three or four) in the assessment by two or more interviewees of each case is selected. These selected complexity elements are described as the most important complexities in each project. The full assessment results are represented as a graph per interviewee and can be seen in Appendix D. Further, the average complexity scores of each interviewee are calculated to get an overview of the complexity dimensions in each project.

Project management approach

Based on the results of the interview, the project management approach for each case is determined. The themes are used to determine the management approach on a certain aspect of the project. All the interviews per case are compared to each other and are represented in Appendix E on different themes formed based on part B of the interview questions.

4.2 Case 1: Project A

4.2.1 General description

It was a development project with a new customer. The project was a step further from the earlier project done by the customer. The project consists of several subsystems (modules). The company was responsible for the definition, concepts, design, manufacturing, procurement, integration, and testing of the specified modules. The customer was responsible for the overall bill of materials of the system, build, and factory acceptance and testing. All engineering development was billed at an hourly rate. The hardware manufacturing was managed on a case-by-case basis, which means that decisions are made based on facts about the current situation, as the time to market was the driving factor. The company had to develop new modules and modify the existing designs made by the customer previously, which includes mechanical, electrical, software, and mechatronics elements. The focus was to use as many subsystems from the previous project as possible.

In this project, there were many subsystems. In one of the subsystems (A), it was decided to use the existing one from the previous project with minimal changes to attain the performance specifications. In another subsystem (B), it was decided to use a modified concept of the subsystem used in the previous project to meet the completion timeline. For subsystem (C), it was a completely new design; the company was responsible for the concept and design. The customer outsourced the manufacturing of the subsystem (C).

According to the customer timeline, the highest priority was given to the design of the subsystem (C). The design of the subsystem (C) was driven by many factors, which made the feasibility and concept phase focused on major unknowns. Also, the design of the subsystem (C) was dependent on interfaces with other modules. Therefore, the focus for (C) was on defining and freezing module requirements and interfaces to finalize the design. For subsystem (D), the customer was responsible for its alignment with the main system. In the concept phase of the project, the positioning of all components was the biggest challenge. The design of one of the components required selection between two options. The option, which had a high performance with low project risk, was selected. One disadvantage of this choice was that it had multiple inputs and output components, which made it complex, but had low project risk because it was made by a limited set of mechanical components compared to other choices. The customer-owned the software and asked the company to develop it further to control the modules and to achieve more throughput. The software was developed on the basis that it interacted with customer's software and was able to do the required system and motion control. To summarise the above, some of the subsystems were used from the previous project, and some were developed. Modules designed by the company were manufactured at the other sites of the company, and one of the modules in the project was manufactured in China. The company held the design responsibility, designed two modules fully, and redesigned some modules from the previous project. The customer designed four modules fully, and the customer based on the inputs of the company designed one module.

The project consisted of a multidisciplinary team with engineers from the following disciplines: Electronics, Mechanics, System Architecture, Motion Control, Software, and Manufacturing. Since there was a lack of skilled people, the company hired external consultants to review their designs and were involved from the start of the project.

The project included both software and hardware development. The PGP of the company was used as the standard process of development in the project. The product was not to be produced in volume series, and therefore the PGP pilot phase was not applicable. The PGP is used to identify and track deliverables for each phase. The Partner web was used to share the documents between the company and the customer. The waterfall approach was used to manage HW development, and for software development, the Scrum method was used.

External dependencies in the project were the small team from the customer side for providing answers to all the design-related questions and choices. Secondly, the manufacturing of the deliverables designed by the company was done at a different location. Thirdly, several critical component suppliers were involved; therefore, the company had to check the lead-time before making a design choice. The probable risk identified early in this project was that technical requirements were subject to frequent changes by the customer.

Table 4.2 describes the role and information about the respondents of the interviews from project A.

Table 4.2 Interviewees and team of project A

Code	Role	Experience	Responsibility	Team size
PL1	Integral project leader	more than 20 years of experience	Whole project	15 to 20 members
HW1	Lead for HW development and a designer in the team.	18 years of experience as a designer but this is the first project as the lead	Mechanical development of the whole project	2 members
SW1	Lead for the SW development team and software architect	more than ten years of experience	Software development of the whole project	3 members

4.2.2 Project complexity

Table 4.2 presents all the complexity elements selected from project A. Based on the complexity assessment done by all the three interviewees; it is seen that the PL1 marks quite a lot of complexities high compared to what is realized by HW1 and SW1. It is observed that there are no similar complexity elements between HW and SW and no similar complexity elements observed by all of the interviewees.

Table 4.4 represents the average complexity score for project A. From the complexity averages; it is clear that the HW development of the project is more complex than SW development in case of technical, organizational, and external complexity. The overall complexity average marked by the interviewee PL1 is almost double the complexity average marked by the interviewees (HW1 and SW1).

Table 4.3 Complexity elements from project A

Complexity elements	PL1	HW1	SW1
Uncertainties in scope	✓		✓
Strict quality requirements	✓	✓	
High number of tasks	✓		✓
Dependencies between tasks	✓	✓	
Involvement of different technical disciplines	✓		✓
High project schedule drive	✓	✓	
Lack of resources and skill availability	✓	✓	
Number of different nationalities	✓	✓	
Involvement of different time zones	✓	✓	
Incompatibility between different project management methods	✓		✓
Level of competition	✓		✓
External risks	✓		✓

Table 4.4 Complexity average for project A

Complexity average	PL1	HW1	SW1
Technical	3.1	1.7	1.4
Organisational	2.5	1.5	0.9
External	2.2	0.8	0.6

Scale: None (0) – little (1) – some (3) – substantial (3) – very much (4)

Based on the interviews, it could be said that PL1 gives an overall view that all the complexity elements are managed well. Whereas according to HW1, overall complexities were not managed well; all were a problem except “number of different nationalities.” The remark pointed out by HW1 is the need for more resources in the project. According to interviewee SW1, “Unclarity of project goals,” “Uncertainties in scope,” and “Incompatibility between different PM methods/tools” are the complexity elements that were not managed even when using the Scrum method. Also, SW1 remarks that the “level of competition” was a problem because of the use of new technology, which took much time to learn. This affected the project because there was already high pressure on time.

4.2.3 Project management approach

The answers of the interviewees of project A are analyzed and summarised per theme. A brief discussion and observation of the management approach per theme is provided below.

HW/SW development

According to all the interviewees, the WOW chosen for HW and SW is based on the general way of working of the company. Other reasons mentioned by the interviewees are that according to PL1, HW cannot be developed iteratively and nor does he have a positive experience with agile for HW, but this reason is questionable as the interviewee does not know much about Scrum and it is WOW based on interview. The reason given by another interviewee (HW1) points out that there is a lack of availability of resources in the company to do the work in parallel. The reason given by the SW1 is

that to follow Scrum fully, full commitment is required and not do anything else, and this is currently not possible in the company.

Project planning

The interview answers show that the project is planned using the PGP of the company and based on efforts. PL1 mainly does the planning, and others follow it. The answer of the interviewees (PL1 and HW1) shows that the scope was not frozen in the project. SW development team made their estimates, but there were also underestimations due to unclear requirements from the customer and uncorrelated work with other teams in the project, as mentioned by SW1.

It is clear from all answers that planning was delayed in the project. Overall the delay in the project was caused to improve the quality at the end, which shows that the quality was the highest priority. The reason for the delay, according to SW1, is due to the underestimation of the requirements because of unclear requirements and problems with the suppliers. Whereas for PL1, it was because of the dependency with the customer as they were not ready at the required time.

Time, Cost, Quality, and Scope

Based on the answers, it is clear that the core team with the PL1 was involved in defining the project promises. Since the customer had a fixed deadline, the project had a fixed schedule. There also seems to be a problem in the trade-off between time and quality according to the answers given by all the interviewees. The interview answers about scope show that there were changes in the scope, and it was not fixed. The answer given by the HW1 tells that an interface was missed at the start of the project but added later in the planning. In the SW team, the inputs were taken from the team and customer to define the scope, but the customers did not have all the requirements to give inputs.

Project goals and deliverables

The answer of PL1 and HW1 indicate that the project goals and deliverables were defined with the customer from the initial phases of the project, which shows good involvement of the customer to define the goals and deliverables. Whereas, in SW development, deliverables are identified by the structure of SW, and it is defined with the team.

Tasks definition

For the overall project, tasks were defined by the PL1 with the core team based on WBS through SOW and PMP. PL1 prioritizes the tasks and assigns them based on the right competencies, and this is done at the start of the project. However, for the SW team, the tasks were defined by the Scrum WOW and picked by the team. Also, the SW team used Jira to track the progress, but the problem noticed is that the team members did not update it regularly.

Requirements

Overall, the answers of the interviewees show that the requirements were changing in this project, and according to PL1, this is expected from a prototype project. According to HW1 and SW1, the initial requirements from the customer were not complete and therefore took much time to discuss. According to HW1, changes in requirements after the design phase could not be incorporated, as it

was too late. In the case of unclear requirements, the interviewees indicate the back and forth approach is followed (making the assumptions from the requirements and then customers reviewing them). According to PL1, the use of the team center has helped share all the documents and for the teams to know the requirements. Besides, the use of a one-room approach has been helpful for discussions with the team. In SW development, since the requirements were not complete, the planning was based on the deliverables, and the customer accepted changes until it was too late, as mentioned by SW1.

Customer engagement

There was constant and overall good interaction with the customer. PL1 mainly did the communication with the customer. Since the customer was from a different country, there were fewer face-to-face meetings, but this was not observed as a problem, according to HW1 and SW1. For interviewees, it is more about getting to know and interacting with the customer to avoid problems. According to the interviewees, the progress was shown to the customer only when something was ready. It can also be seen that the team had no contact with the customer. The team communicated the progress to PL1, who was in contact with the customer.

Team engagement

The answers of the interviewees show though it was a big team, a one-room approach helped the team to engage well. Also, other than meetings, there was much of informal communication that took place within the team. Though there were good interactions within the teams, one problem indicated by the SW1 is that a team does not know about the progress of other teams in the project, which should be improved. Within the SW teams, Jira was used to visualize the progress of the team.

Project evaluation

It is clear from the interview that though there were evaluations conducted for the whole team by the PL1, there were no separate team evaluations done. Even though the SW team was following the Scrum method, they still did not do any retrospectives within the team.

Interface

The PL1 managed the interface through the integration schedule. The interviewees (PL1 and SW1) commonly mentioned that it is difficult to manage the milestones when different management approaches exist in the same project; therefore, consider it as a problem. Overall, the interfaces could be better managed if the communication with other teams is maintained well from the start mentioned by SW1. Also, according to SW1, the main challenge is a lack of understanding of the approach among the team, which makes it difficult to manage, and other disciplines are afraid of micromanagement, therefore, refrain from using Scrum.

Current way of working

Based on the advantages mentioned by the interviewees, it shows that following PGP, one-room approach, and having daily stand-up was helpful. In contrast, the disadvantages mentioned by the interviewees are lack of resources, lack of clear understanding of requirements, and that long duration is difficult for the members to focus. According to PL1, the overall approach could be

improved by having faster testing methods like 3-D models, whereas, according to HW1, the approach could be improved by setting the right priorities between the quality and time. However, according to SW1, the overall approach could be improved by making the progress visualization for the team and by educating other teams about Scrum.

Scrum

There is limited knowledge observed in HW1 and PL1 about Scrum, but HW1 is open to learning about Scrum, whereas PL1 is not interested in applying scum for HW except for daily stand-ups. According to SW1, everything can be applied if the difference between HW and SW is considered. This shows that except for SW1, nobody knows about Scrum fully.

4.3 Case 2: Project B

4.3.1 General description

It was a development project with an existing client. It was a redesign of a previous existing product (B) to realize new features as the customer required higher productivity by increasing the throughput. The customer, in principle, gave the requirements, and the company made the design. The project needed continuous alignment with the customer due to time constraints. The constraints were related to the availability of proto test time, planning with the customer's factory for the proto and pilot phase, planning with SW from the customer for integration and release, availability of resources within the company for performing the project tasks, and progress of suppliers. The cost of the project was based on the efforts estimate. The intended throughput increase was essential for the success of the project. The team agreed on a systematic approach for implementing the improvements. A conscious trade of time vs quality was made for the introduction of the packages.

It was in the scope of the project to fix the issues caused due to changes in the project that affect the reliability, availability, and throughput. Also, repair the damage in availability due to the increase in throughput. The project consisted of a multidisciplinary team with engineers from the following disciplines: Electronics, Mechanical design, Software, Mechatronics, and Manufacturing.

The project included both software and hardware development. The company used PGP as the standard of execution of the project. The different phases of the project were running in parallel, and within these phases, the work packages were defined and executed. The communication in the project with the core team of the customer is via partner web for sharing documents, deliverables, progress reports. The project management way of working for software development was based on Scrum. For hardware development, the waterfall approach was used for planning, and partly Scrum was used for alignment with teams. External dependencies in the project that had a possible influence on the project plan were first because of design, as some of it was outsourced. Secondly, because testing is done on the customer proto, this affected time. Thirdly, because of different suppliers, this led to consider of lead-time for deliveries. Table 4.5 describes the role and information of the respondents of the interviews from project B.

Table 4.5 Interviewees and team of project B

Code	Role	Experience	Responsibility	Team size
PL2	Integral project leader	Two years of experience as the leader for SW development but 1st multidisciplinary project	Whole project	15 to 20 members
HW2	System architect	Three years of experience as an architect but this is the most complex project	Technical details of the end product	12 members
SW2	SW developer	Six years of experience	software development	6 members

4.3.2 Project complexity

Based on the complexity assessment done by all the three interviewees, it can be observed that the Involvement of different technical disciplines, High project schedule drive, Lack of resource & skills availability are the complexity element marked high by all of the interviewees. In project B, SW2 observes no external complexity element for SW development. Table 4.6 shows the selected complexity elements in project B.

Table 4.6 Complexity elements in project B

Complexity elements	PL2	HW2	SW2
High number of project goals	✓		✓
High number of tasks	✓	✓	
Involvement of different technical disciplines	✓	✓	✓
High project schedule drive	✓	✓	✓
Lack of Resource & Skills availability	✓	✓	✓
Lack of Experience with parties involved	✓	✓	
Number of different nationalities		✓	✓

From the complexity averages, it is clear that SW development is technically more complex than the HW development of the project. However, HW development is more complex than the SW development in case of organizational and external complexity. Also, the overall complexity seen by HW2 and SW2 is more than PL2. Table 4.7 represents the average complexity score for project B.

Table 4.7 Complexity average for project B

Complexity average	PL2	HW2	SW2
Technical	1.8	1.9	2.1
Organisational	1.2	1.9	1.5
External	0.5	1.3	0.8

Scale: None (0) – little (1) – some (3) – substantial (3) – very much (4)

Based on the interview, daily Scrum helped in handling the "involvement of different technical disciplines" (Complexity element) for PL2 and SW2. According to the interviewee, it would be better to involve the factory side of the company from the start of the project. According to HW2, the non-functional requirements were focused more at the end, and this was not a problem as the customer also worked similarly. Other remarks made by HW2 are that the customer was controlling the project, and there were problems with suppliers and new suppliers are not approached because of the internal politics of the company (PL2). Scrum method as such did not influence the time but just gave an overall picture of the progress in the project (SW2).

4.3.3 Project management approach

The answers of the interviewees of project B are analysed and summarised per theme. A brief discussion and observation on the management approach per theme is provided below.

HW/SW development

The overall Scrum WOW was chosen for both HW and SW, but inside the HW development team, the development and planning were based on the waterfall approach. The answers of the interviewees show that it is the general WOW for both HW and SW to be developed using a waterfall and Scrum respectively and is not a choice. According to SW2, the Scrum used for SW team had to be adapted because of different WOW of the customer. This shows that customer's WOW affects the company's WOW when there is a dependency of work. According to PL2, working in Scrum helps to reduce and manage the risks in the project. The communication was streamlined by making the HW and SW work together in one location so they can discuss everything and not go in different directions.

Project planning

The planning was based on PGP and Scrum practices (daily stand up, backlogs). According to HW2, it was simultaneous planning of phases because doing sequential will affect time, and there was a tight schedule in the project; therefore, risks were taken, and the phase was kept open. The decisions related to outsourcing were based on the available competencies. For the SW development, the planning was based on the WBS made by the team, and if something did not fit in the desired timeline, then it was further negotiated with the customer.

Time, Cost, Quality, and Scope

Scope, time, and quality were fixed in this project. The planning was done based on the rough estimate discussed with systems architects and lead designers. The team was not involved in making the rough estimates but later involved in splitting the tasks into small ones. From the specifications given by the customer, the requirements and scope were decided. The priority for the HW2 and SW2 was time and quality, whereas the priority for the PL2 was scope and time. Overall time was the highest priority. Only the PL2 indicated that the time in the project was overspend, whereas other interviewees (HW2 and SW2) indicate the project was completed on time.

Project goals and deliverables

The goal and deliverables are made from the statement of work (SOW), which is reviewed by the management team of the company and based on the PGP. The team is not involved in defining, and

they get to know only during the kick-off of the project. However, in SW development, the team with the SW lead defines the goals and deliverables in the sprint based on WBS.

Task definition

The PL2 and key members of the project define all the tasks. The WBS is discussed with the customer, but the customer is not involved in making the WBS. The resources and time required are defined by effort-based estimation made by the lead engineers then discussed with other engineers. All the interviewees mention that the tasks are distributed based on the competencies in both HW and SW development.

Requirements

The requirements were co-developed in close collaboration with the customer and more in one to one discussions. The key players review the requirements and set priorities, and then the team gets to know during the kick-off. Overall, in the project, there were no changing requirements from the customer and requirements were clear from the start, whereas the timeline and order of introduction of certain packages changed from the company's side according to the interviewees (PL2 and HW2). Also, for SW development, the priority of requirements is based on the customer's wish list.

Customer engagement

There was constant and regular interaction with the customer. Interviewees (PL2 and HW2) indicated that progress in the project is visualised through presentations only when they had results to show, and the team was not part of the progress meetings. Besides, there were no changes from the customer side, and in fact, they were cooperative with setbacks from the company's side.

Team engagement

According to all the interviewees, there was daily interaction with the team (daily stand-ups or informal interaction). The progress in the project is visualised through presentations, Jira, and a dashboard of things to do and done was maintained, but this was done only for the SW team. The problem was that the team did not know the progress of other teams. However, as indicated by SW2, Jira did not work for them as not everyone knows to use it, and because of more tasks, it was not feasible for them to create a backlog.

Project evaluation

PL2 conducted the evaluation meetings only with SW teams. This also shows that there was no project evaluation conducted with the overall team or with other teams.

Interface

PL2 managed the interface, and the use of one room approach helped the teams to discuss and work effectively. According to PL2, it was a challenge to have different management approaches in the same project, as it was difficult to match the milestones between teams. According to the SW2, testing is done at the customer site, as the testing facilities in the company were not sufficient, which shows it was difficult and challenging to work without testing and this overall affects the time.

Current way of working

Based on the interviews it is shown that the current management approach and using PGP helps to focus on the to-do whereas, for SW development, daily stand up meetings helped to see the dependencies in the teams. The disadvantages mentioned by all the interviewees show that long-duration makes it difficult for the team to focus, teams tend to move the task to the next sprint without focusing on completing it, and also PGP is very formal which makes skipping difficult. The interview answers show that the Scrum was effective in managing the complexities. Nevertheless, the current approach could be improved by making the teams understand the value and by bringing discipline in the team to fill data (PL2). SW2 suggests that the sprint length should be longer in the designing phase and comparatively shorter in the testing phase because they could not deliver much in the initial phase, as a long time was required. Also, the interviewee mentions that Scrum could be improved by having more flexibility, scalability, and having less fixed rules, and planning poker does not work since everyone works separately. They do not know about somebody else's task. Further, SW2 mentions that having a common database to share the knowledge about the work done could be beneficial because if someone leaves the project, it is then easy for the new person to take over by getting the knowledge from there.

Scrum

All the interviewees collectively think that daily scrum meetings can be applied and further investigation is needed on what else can be adapted. Though for PL2 and SW2 collectively think Scrum will work for HW, this could be because both have the background and knowledge of working in SW and Scrum. Whereas HW2 does not know about Scrum completely, still thinks Scrum will not work for HW development.

4.4 Case 3: Project C

4.4.1 General description

Project C is an installed base project, managing small changes or improvements in already released systems that are in the field. The project is not about developing a new product but about maintaining the product at the customer site. The project is with an existing customer and involves many stakeholders. The customer does not give specifications but gives detailed requests to solve an issue in the delivered product by the company. Therefore, when there is an issue with the product at any of the sites of the customer, the customer escalates the issue to the company by creating an issue request with a description called issue resolution. The customer updates the issue resolution in the database created by the customer. The issues go through multiple phases from causes to the solution, and the planning of each issue resolution is based on the priority of the customer. The customer gives the overall budget for the whole year, and the number of issues that can be solved within that budget depends on the complexity of each issue.

The project consisted of a small multidisciplinary team with engineers from the following disciplines: Electronics, Mechanical design, Software, and Mechatronics. The way of working is dependent on the customer and is a strict process. The solution may require the development of HW or SW or both. The team handles the issue; the project leader assigns the representative for the issue to solve it

based on their availability. Table 4.8 describes the role and information about the respondents of the interviews from the project C.

Table 4.8 Interviewees and team of project C

Code	Role	Experience	Responsibility	Team size
PL3	Project leader	Two years of experience	Take input from the customers the plans and manages issues in each quarter. And Later gives feedback to the customer	Up to 10 members
HW3	Mechanical lead designer/engineer	Seven years of experience as a designer	working with hardware depending on the issues	4 to 5 members
SW3	Software project leader	Five years of experience and ten years of experience working with agile	Responsibility of Scrum master and product owner	5 members

4.4.2 Project complexity

Bases on the complexity assessment done by all the three interviewees, it is seen that PL3 has marked very few complexities high. Based on the complexity elements marked high by all the interviewees, it is observed that all of them mark “High number of tasks” high. Also, other than the similarity mentioned above, “High number of project goals” and “interfaces between diff disciplines” are similar complexity elements between HW and SW development. All the complexities selected from project C are represented in Table 4.9.

Table 4.9 Complexity elements in project C

Complexity elements	PL3	HW3	SW3
High number of project goals		✓	✓
High number of tasks	✓	✓	✓
Involvement of different technical disciplines	✓		✓
Interfaces between different disciplines		✓	✓

From the complexity averages, the SW development is technically more complex than the HW development of the project. However, HW development is more complex in the case of organizational and external complexity. Besides, the overall complexity average of HW3 and SW3 is higher than PL3. Table 4.10 represents the average complexity score for project C.

Table 4.10 Complexity average for project C

Complexity average	PL3	HW3	SW3
Technical	1.6	1.9	2.2
Organisational	1.1	1.8	1.5
External	0.8	1.9	1.5

Scale: None (0) – little (1) – some (3) – substantial (3) – very much (4)

Based on the interviews, PL3 remarks that weekly meetings should be further improved as the interviewee has to align regularly with many external stakeholders. However, according to HW3, there were many problems related to communication within the team, which needs to be improved; other than that, the HW3 remarks that the company and the factory of the company have to work together from the start. According to SW3, the Scrum method has been overall helpful to manage the complexities, but some problems found in Scrum are that some members do not like using Jira, as they do not see an advantage in it. Also mentions that using Excel is tedious for filling the data.

4.4.3 Project management approach

The answers of the interviewees of project C are analysed and summarised per theme. A brief discussion and observation of the management approach per theme is provided below.

HW/SW development

The interview answers show that for HW development waterfall approach was used, and the customer's WOW has an influence on the development process. There were weekly alignment meetings conducted with the team by PL3, which the interviewee (PL3) considers as Scrum WOW. Also, the customer in this project is involved from the start to gain control over the process and decides whether or to proceed with any solution. The SW development, on the other hand, was developed fully by Scrum, and it was the initiative of the company.

Project planning

Based on the interview answers, the customer gives the wish list for which issues to look at first and then, based on that, PL3 makes the plan and assigns a member to be in-charge for the issue. According to HW3, making estimates is difficult because it is not one issue to work on at a time, and each issue is like one project. When the issues involve multiple disciplines, then the planning is done together, estimates are discussed, and the dependencies are known at the kick-off. For the SW development, the whole planning is done in Jira (from deciding on the tasks in sprints to backlog), and tasks are assigned based on competency. According to SW3, Jira is used because it helps to spread the tasks across the sprints, and this allows the interviewee to predict by when an issue will be solved.

All the interviewees have a positive opinion about the success of the project. However, the answers of some interviewees (PL3 and HW3) point out that though the project is successful in the end, there are problems with the communication within teams and the teams do not know about the progress of other members.

Time, Cost, Quality, and Scope

From the answers of the interviewees, it is clear that the customer defines the deadline and quarterly plans are made based on it. However, the customer was fine with the delays, which shows that though the time is fixed, it is not the highest priority. The budget for the project is fixed for the year, and PL3 defines the cost for each issue based on experience. The highest priority is given to the quality of the work, and the customers do ask for the guarantee for the work done. Also, the scope of the work keeps changing as it is based on the issue, and it can be realized only after working on the issue. According to HW3, the estimates are done by the representative assigned for the issue

individually, but sometimes the PL3 is involved when the issues are bigger and if the issue involves other disciplines, then even they are involved.

Project goals and deliverables

According to PL3 and HW3, the goal is to solve the issues. The PL3 monitors the deliverable and all the deliverables are maintained in an Excel sheet. The problem of maintaining an excel sheet is that not all the members fill it. So the team would not know about all the deliverables. In the SW team, the goal and deliverables are clearly defined for each sprint. The WBS made by the representatives assigned for the issue defines the deliverables, and the lead engineers review it. The SW3 also maintains a progress document, which is sent to all the stakeholders, so all of them know about the status of all the issues.

Task definition

The tasks are defined based on WBS and are assigned based on the competences. The estimates for an issue are made before starting to work on it. The team knows about all the tasks that are going on during the weekly alignment meetings. In SW teams, the team discusses the estimates at the retrospective meetings also to keep a check.

Requirements

PL3 defines and prioritizes the requirements. One problem addressed by all the interviewees is that there are sometimes unclear requirements; in that case, it is communicated with the customer to get it cleared. Also, the team is in contact with the customer directly. There are sometimes changes in requirements from the customer; in that case, PL3 does not accept once the team starts working on it (after the kick-off). However, according to SW3, the SW development team is flexible with the changes in the requirement. Also, there is a weekly discussion with the customer by the interviewee (SW3) to check the status of the requirements and performance of the team.

Customer engagement

Weekly the PL3, key members, and the team discuss the progress in the issue in which the customer is not a part. Also, there are weekly meetings to discuss new issues added to the list. This shows there is frequent and good interaction with the customer. The team does not know about the progress of the overall issues and communicates to the customer through the PL3, according to the interviewees (PL3 and HW3).

Team engagement

There are weekly meetings conducted by the PL3 with every discipline separately. Also, there are many informal discussions within the teams. The teams engage with other teams only when there is a need. However, in SW, the member assigned to an issue is constantly involved from the start, so it is easy to communicate with the customer directly when needed, according to SW3.

Project evaluation

There are no regular evaluation meetings conducted for the overall team or inside the HW teams. In SW development, teams do retrospective/review meetings after every sprint.

Interface

There are alignment meetings and kick-off meetings to give clear details about the roles and interfacing with other teams. The interviewee (PL3) manages the interfaces with other teams, but if there are more interfaces involved, then the teams take the effort to manage it. According to SW3, it is a challenge to have different management approaches as there is a difference in expectations. The interviewee (SW3) suggests that it is better to follow the same procedure throughout as this will make the deliverable sink in well.

Current way of working

According to PL3, the overall planning is flexible but suggests that Scrum could be used to make an overview of the planning visible to all the team members and for better estimates. Further, the interviewee (PL3) suggests that the management approach can be improved if the teams have a common worklist, whereas HW3 points out that the use of a team center is initiated to improve the sharing of documents and to align with other members. According to SW3, some people do not like Scrum because they think it is micromanagement. Moreover, such a problem can be solved if the team members are given more responsibilities, which would make them feel empowered, according to SW3.

Scrum

The PL3 does not know much about Scrum, and the SW3 does not know much about HW, which makes it difficult for both of them to suggest Scrum for HW development. However, the PL3 thinks that stand up meetings could be adopted but does not require it to be daily. Since the SW development team already have daily stand up meetings, it is better to have separate stand up meetings for others (PL3). Scrum method would not work for HW development, as more time is needed to fully investigate and fix the issue in HW development (HW3). Whereas SW3 suggests that Scrum might be helpful for HW development, as it also has to be tested, and not everything can be planned before, which makes the Scrum suitable for HW.

4.5 Conclusion of the case study

All projects (A, B &C) were selected based on the case selection criteria. The data for each project was collected by three methods: case documents, complexity assessments, and semi-structured interviews. The project documents were interpreted to give basic information about each project.

Based on the theoretical framework, themes in the management approach were selected. These themes were formulated as questions for the semi-structured interviews. All the interviewees were questioned to gather information on the selected themes. Further, the results of complexity assessment from the TOE framework (Bosch-Rekveldt et al., 2011) show the complexity characteristics. Overall, the Build-to-Spec projects (A & B) tend to follow the management approach and development process based on the general way of working of the company. For the Installed Base project (C), the customer influences the overall way of working of the project. Both project A and B (Build-to-Spec) follow a plan-driven approach for HW development where the project leader does the planning with the key members based on the efforts estimates, assigns tasks mainly based on the competencies and finally conducts an evaluation meeting to know the lesson learned at the end of the project. Also, the project leader manages the interface between HW and SW development. However, the SW development in all projects (A, B & C) follows a Scrum method.

In projects A & B, the project leader maintains close contact with the customer and shows progress when something is ready. However, for the Installed Base project (C), the project leader maintains close contact with the customer and shows progress to the customer every week. All the projects maintained a constant and informal interaction within the team. Also, in both the projects A & B, the project leader took the initiative to use a team center and one-room approach for the team to engage closely. Further, cross-case analysis is done in the following chapter 5.

5 Cross-case analysis

In this chapter, the three cases mentioned in chapter 4 are compared with each other. Section 5.1, shows the comparison between three cases based on their project characteristics, project complexity, and project management approach from the analysis. Section 5.2 draws the suggestions of scrum elements that apply to the complexity elements and problems addressed in the cases. Section 5.3 discusses the conclusion of the cross-case analysis.

5.1 Comparison between the 3 cases

5.1.1 Project characteristics

Table 5.1 is made based on the general description of all three cases to understand the differences in the projects. Project A and B are similar as both are development projects, significant in terms of team size (15 to 20 members), and has high complexity in the product design, compared to project C. Project C is a comparatively small project (10 members), it is more about doing small improvements or changes to the already delivered product. In the case of project A, it was a completely new customer, and for project B and C, it was a well-known customer of the company. All three projects have high external dependencies. Table 5.1 gives an overview of the project characteristics of all three cases.

Table 5.1 Project characteristics (own illustration)

	Project characteristics		
	Project A	**Project B**	**Project C**
Type of the project	Build to Spec	Build to Spec	Installed-base
Product	New product	Improvement of the previously existing product of the company	Small changes or improvements in already released systems.
Customer	New and from a different country	Well known customer	Well Known customer
Product design	Challenging and highly complex	There is a legacy, therefore, not much freedom to do change. Also, learnings from the previous project could be applied. Therefore the risks were known	Complex and many issues to work at the same time. Each issue is like one project
Location	Three sites were involved in the company side, one for designing, and two for manufacturing (one of them in another country)	No sites, only T&D, and manufacturing department of the company involved	No sites involved from the company side but the customer side involved stakeholders from various locations
External dependency	Many due to suppliers, different manufacturing sites and shared responsibilities with the customer	Outsourcing of some design components, testing at the customer location and involvement of different suppliers	Many stakeholders involved at the customer side
Team	Large Multi-disciplinary	Large Multi-disciplinary	Small multidisciplinary
Time	Fixed deadline and high pressure	Highest priority as high pressure to be in time	Fixed but not highest priority
Cost	Flexible	Flexible	Flexible for each issue, the budget is fixed for a year

5.1.2 Project complexity

The average complexity score indicates that project B and C have an almost similar set of scores marked by all the interviewees for all Technical, Organisational, and External complexity. This makes projects B and C similar in terms of complexity. The average complexity score indicates that in both projects B and C, the technical complexity of SW development is higher, whereas for project A technical complexity of HW development is higher than the technical complexity of SW development. The complexity averages of the PL1 for Technical, Organisational, and External elements in project A are remarkably higher than the complexity averages of PL2 and PL3 in their respective projects, which shows that PL1 has a different perspective.

Besides, the complexity assessment also helps in determining specific complexities that affect the projects of the company most. The number of elements that are marked high (3 or 4) by more than two interviewees is more for project A compared to project B and C. Therefore, it can be said that project A has more complex elements involved than the other two projects.

When looking at the complexities selected from all the cases, “High number of tasks” and “involvement of different technical disciples” were present in all the cases. Since projects A and B are Built-to-Spec projects, it is seen that it has four similar complexity elements, namely “High number of tasks,” “involvement of different technical disciples,” “High project schedule drive,” and “Lack of resources and skill availability.” Table 5.2 summarises the list of all selected complexities from the three cases. This list of complexity elements does not mean that all the complexity elements are present in all projects; it summarises the complexity elements from all the projects. A tick in figure 5.2 represents the presence of the complexity element in the respective case.

Table 5.2 Complexity elements in the projects

Complexities	Case 1	Case 2	Case 3
1. Uncertainties in scope	✓		
2. Strict quality requirements	✓		
3. High number of project goals		✓	✓
4. High number of tasks	✓	✓	✓
5. Dependencies between tasks	✓		
6. Involvement of different technical disciplines	✓	✓	✓
7. High project schedule drive	✓	✓	
8. Lack of resources and skill availability	✓	✓	
9. Lack of Experience with parties involved		✓	
10. Interfaces between different disciplines			✓
11. Number of different nationalities	✓	✓	
12. Involvement of different time zones	✓		
13. Incompatibility between different project management methods	✓		
14. Level of competition	✓		
15. External risks	✓		

5.1.3 Project management approach

The management of both the project A and B follows a fixed plan-driven management approach compared to project C. All three projects A, B & C have the highest priority to quality but project A and B also have a high priority on time to achieve the quality whereas project C can accept delays and therefore has the flexibility of time.

When analyzing project A, it is seen that the scope of the project had some changes and was not fixed. Inputs were taken from the team and the customer, it is seen that customer in the start supplied insufficient and unclear requirements which led to changes in scope, but this is not the case with project B. Also in project A, changes in requirements were expected, but this was restricted after the design phase for HW development. In the case of project B, the requirements did not change from the customer side. Project A seems to show of lack of customer involvement as the interviewee (SW1) mentions that there was a point in the project when there was no contact with the customer for a long time. However, since project C is an installed base project, it is expected by the interviewees to have scope changes and changing requirements. When analyzing project C, it is seen that all deliverables are maintained through the excel sheet, but the problem observed is that not all members fill the details. Also, since many tasks are involved, it makes it difficult to estimate and prioritize the tasks.

A commonly addressed problem in both the Build-to-Spec projects (A & B) was the problem with suppliers that affected the project schedule, which shows that these projects have a high dependency on their external stakeholders. The next addressed problem was that the long duration makes it difficult for the team to focus. This problem shows that these projects follow a very long schedule and phases of work that makes it difficult for team members to focus. Also, the interviewees in the projects A & B commonly address the lack of availability of members as a problem. Overall, in all the projects, it was a challenge to match the milestones when there is a different management approach involved in the same project.

The SW development of the project of A & B uses Jira but was not regularly updated, which shows a lack of knowledge and coaching about agile in the company. Also commonly mentioned by the interviewees that SW development teams do not know the progress of other teams in the project, which is a problem as their work is sometimes dependent on other teams. In project B, the lack of flexibility in the sprint length and moving tasks to the next sprint without focusing on completing is observed as a problem. The problems addressed in the Scrum method will be discussed in the discussion chapter (refer to chapter 7).

It is also seen that interviewees from the HW development are not fully open to apply Scrum for HW development irrespective of their knowledge about Scrum. Besides, all the interviewees, in general, suggest that daily stand up meetings could work for HW development, which shows that either they are unaware of other Scrum practices or they could be not open for other practices. Based on the analysis of the interviews, problems encountered during the management of projects are summarised as a list and presented in Table 5.3. This list of problems does not mean that all the problems are present in all projects; it summarises all the problems from all the projects. A tick in figure 5.3 represents the presence of the problem in the respective case. Also, these are only the problems observed in the traditional management of HW development.

Table 5.3 All problems in the cases

Problems	Case 1	Case 2	Case 3
1. Scope changes	✓		✓
2. Unclear requirements	✓		✓
3. Changing requirements			✓
4. Lack of availability of members		✓	
5. Lack of customer involvement	✓		
6. Non-alignment of milestones between HW and SW team	✓		✓
7. High pressure of time	✓	✓	
8. Problems with suppliers	✓	✓	
9. Unclear priority between tasks		✓	✓
10. Insufficient requirements	✓		

5.2 Fitting the Scrum elements based on the project complexity and problems identified

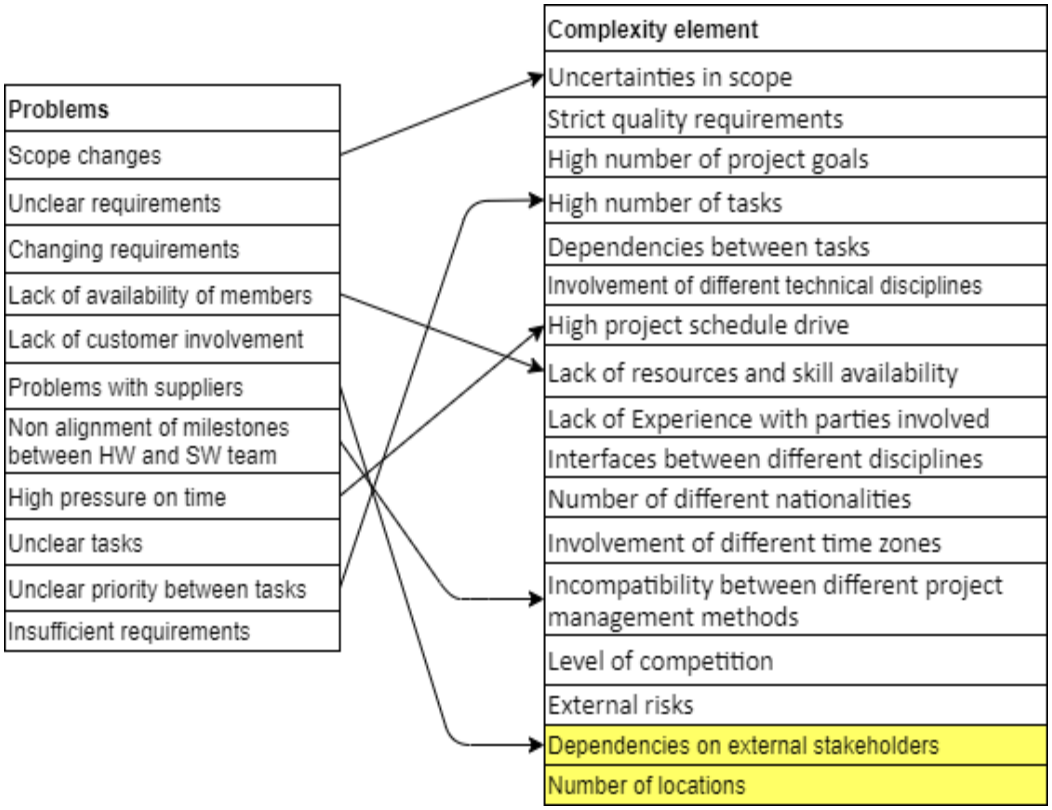


Figure 14 Relating problems to the complexity elements

For choosing the right elements from Scrum practices, a list of all complexity elements and problems identified is made based on the complexity assessment and case study. However based on the project characteristics studied "dependency on external stakeholders" and "Number of locations," are the new complexity elements added and represented in the yellow box in Figure 14. Though the interviewees did not access them in the complexity assessment, they are, however, identified as common project characteristics, therefore, added to Table 5.2. However, when comparing Tables 5.2 and 5.3, it is seen that some complexity elements can be identified as the direct cause of some of the

problems (illustrated by Figure 14); “scope changes” is related to the “uncertainty in scope,” “Lack of availability of members” is related to “lack of resources,” “non-alignment of milestones” is because of “incompatibility between the management approaches,” “high pressure of time” is because of “high schedule drive” in the project, “problems with suppliers” is because of “dependency on external stakeholders” and lastly “unclear priority of tasks” is because of “high number of tasks” in the project. Therefore, those problems are not added to the final list in Table 5.4. For other problems observed in the project, “unclear tasks,” “insufficient requirements,” “changing requirements,” “unclear requirements,” and “lack of customer involvement,” no complexity element was found as a cause; therefore, these are added to the list directly. The selected complexity elements and problems encountered are combined as a list and are represented by Table 5.4.

Table 5.4 List of complexity elements and problems

Uncertainties in scope	Lack of resources and skill availability
Strict quality requirements	Lack of Experience with parties involved
Unclear requirements	Interfaces between different disciplines
High number of project goals	Number of different nationalities
High number of tasks	Involvement of different time zones
Changing requirements	Incompatibility between different project management methods
Dependencies between tasks	Level of competition
Lack of customer involvement	External risks
Involvement of different technical disciplines	Dependency on external stakeholders
High project schedule drive	Insufficient requirements
Number of locations	Project duration

When comparing each element in Table 5.4 to the theoretical framework, the following suggestions on the most promising Scrum elements for each element of Table 5.4 are made.

Uncertainties in scope - To control and reduce the risk of scope creep for HW development projects, using the sprint, Scrum board, and burndown charts can help to see the progress; total time elapsed and track scope fulfilment. This means that if there is a change in scope, it will be visible to the team. With every new scope change during the sprint, it should be communicated within the team through daily stand-ups, discuss its implications, and plan the future sprints accordingly. Since creating a working deliverable is difficult in HW, it is advisable to deliver something that can be discussed during the sprints. The length of the sprint can be flexible, depending on the expected deliverable and the efforts.

Strict quality requirements-When this complexity plays a role in the project; it is essential to be able to do correct trade-offs between time and quality. Only with full team effort, the quality will be achieved. Agile practices are developed to achieve high quality and value in the project. When using sprint planning meetings, the whole team together can come up with solutions that help in solving challenges in the project. It is essential to have the right team, the team should be able to pick their tasks, which will give them a sense of responsibility to complete their task with perfection, and this, in turn, will affect the quality of work done. Also, conducting sprint reviews/ retrospective meetings

with the team will further help to analyze the work done and improve quality. All this can be applied directly irrespective of the difference between HW and SW.

Insufficient requirements and unclear requirements- It is essential to breakdown the product requirements as small as possible for HW. In this way, where more information is needed will be clearly understood. Further, communicating with customers frequently will help to develop the requirements. Using product backlog for requirements will further help to refine and clarify the requirements.

Changing requirements- It is not easy to make changes in HW, and it cannot be redesigned as quickly as SW. However, if the goal is to allow late changes in the requirements in HW, then it is vital to determine early in the process which requirements need to be fixed and which can be allowed to change as this would give the flexibility to the customers. However, all changes need to be analyzed, so firstly, having a product backlog will help to determine which changes are necessary and when to adapt. Also, having sprint reviews involving the customer will help to determine and evaluate the current work and discuss what more changes are needed. Having a sprint will help in controlling the change in requirements as during the sprints, changes are not accepted, and the length of the sprint is flexible to the company. The length is generally the time in which a deliverable could be created by the team that can be discussed.

High number of tasks - With a large number of tasks, it is challenging to create a product backlog even for SW development, which is also mentioned by one of the interviewees (SW2). In this case, the suggestion would be to follow one big task at a time of sprint that is considered achievable. Since HW development has large tasks that cannot be broken down into small ones, in that case, having time-boxed would not work, so the length of the sprint should be considered flexible. It is advisable and also mentioned by one of the interviewees (SW2) to have the sprint length longer in the design phase as it is time-consuming than in later stages of the project.

Lack of customer involvement- The involvement of the customer is very crucial for the success of using the Scrum method in development, be it in HW or SW. Therefore when there is a lack of involvement from the customer side, no Scrum practice can be applied to solve this problem.

High project schedule drive- It is related to projects that have fixed deadlines. In this case, the planning should be based on the fixed scope and fewer uncertainties. Therefore having a product backlog and sprint would help in making a realistic plan that would be achievable also helps in avoiding over-commitment and false promises. The use of sprint retrospectives is also suggested, as this will aid in examining the work done throughout the project. However, it is essential to be in frequent communication with the customer to be aligned in the project always. The use of simplicity (agile principle) should be adapted for HW development, which means it concentrates on those actions that give importance to customer requirements and cutting the extra corners.

High number of project goals- The focus of the project is to achieve all the project goals. Sprints are suggested in achieving all the project goals as it specifies targets and helps the team to work towards them. When there are a high number of goals, it is advisable to have Scrum boards to visualize and keep track of all the goals and development.

Number of locations, Involvement of different time zones, Number of different nationalities- All the three complexity elements can have a possible effect on lack of communication and coordination between team members. In this case, it is advisable to use daily stand-ups (virtually/physically) to maintain close collaboration and resolve language barriers and issues with all members. Daily stand up is applicable irrespective of the difference between HW & SW.

Dependencies between tasks, Involvement of different technical disciplines, Interfaces between different disciplines **-** The product becomes technically complex when different technical disciplines are involved and lead to many interfaces. Also, HW development tends to have high dependencies. When there is a dependency between tasks, then doing the task in parallel is difficult. It is essential to know all the dependencies to manage them. Using Scrum board or other Scrum tools like a burndown chart will help visualize the dependencies, track the project status, and specify the interfaces. Besides, doing daily stand-ups/Scrum meetings is essential, as it will further help team members to communicate clearly about the dependencies with each other, which will improve the correlation between tasks. Further, following sprint retrospective can help to improve the work and resolve issues due to dependencies.

Lack of resources and skill availability, Lack of Experience with parties involved- Scrum method or practices of Scrum cannot be used to manage this complexity directly. However, making the team member pick their tasks can help in identifying what the right skills are needed to do the tasks and where the inadequacy in the project lies.

Incompatibility between different project management methods- Not just one practice of Scrum can resolve this complexity. However, according to the Scrum method, having an overall defined process for a project can help in overall technical excellence and further enhance the agility of an organization.

Dependency on external stakeholders- HW development projects tend to have a high dependency on the external stakeholders, and in that case, time constraint (Time boxed) of the sprint is not feasible. It is required that sprints become flexible, and planning should consider the lead-time and various scenarios. Therefore, the suggestion would be to apply sprints, as it will help with an efficient and realistic plan with proper estimates.

Level of competition- This complexity makes time very crucial in the project that is because of a short time to market and achieve high quality in the end. Agile methods (Scrum) have proven to achieve both. Therefore, with frequent release of deliverables (prioritized backlog, time-boxed sprint, Scrum board) and constant customer feedback, it is possible to achieve it. The length of the sprint is defined based on the deliverable for HW development and not fixed.

External risks- Scrum can help in the identification and monitoring of risks from the early stages of the project. However, the risk caused due to external factors is coming from outside the project. In that case, Scrum does not play a role.

5.3 Conclusion

The cross-case analysis is performed to understand the differences and similarities between the 3 cases. When comparing the complexity elements and problems, it is shown that some complexity elements act as a cause for some of the problems (refer Figure 14). Also, according to project characteristics, two elements were added to the list of complexity elements. A list of complexity elements and problems encountered in the HW development projects are formed. When comparing the final list (refer Table 5.4) to the theoretical framework, it is seen that some set of practices of Scrum can be applied together to manage the respective complexity element/problem (refer section 5.2). However, while fitting the Scrum elements to the list of complexity element and problems in Table 5.4, it is seen that for elements like; Lack of customer involvement, Lack of resources and skill availability, Lack of Experience with parties involved, and External risks, Scrum elements cannot be applied. The summary of the suggestions of Scrum practices is presented in Appendix F.

6 Experts meeting

This chapter discusses the results of the expert meeting. Section 6.1 describes the protocol for the expert meeting. Section 6.2 gives an overview of the meeting with experts and describes the results of the meeting. Lastly, section 6.3 formulates the conclusion of the expert meeting.

6.1 Expert meeting protocol

After the suggestions drawn from the results of the cross-case analysis, a significant question was whether the given suggestions would work for the company to manage the complexities practically. For this purpose, an expert meeting was organised to get their opinion and answer to the question, and further, gather some extra data to support the discussion and conclusion. Three experts from the company participated in the validation meeting; all experts play an active role in the management of HW development projects in the company and were not involved in the case studies. Individual meetings were conducted with each expert. The experts were asked about their roles and years of experience. Table 6.1 presents an overview of the details of the experts.

Table 6.1 Overview of experts for the validation meeting

Name code	Role in the company	Years of experience for the role
E1	Total quality manager	Above 20 years
E2	Program manager	Above 15 years
E3	Operations manager	Above 20 years

During the validation meeting, the experts were introduced to the research context. Various Scrum practices were explained in detail, after which the list of Scrum practices as a suggestion to manage the respective complexity elements and problems (refer to Appendix F) was shown. Later the experts were asked the question about the practicality of the suggestions proposed for the company. It was a discussion between the researcher and the expert on each set of suggestions. The discussions are used to determine the applicability of the suggestions.

6.2 Results and analysis of experts validation

The results from the expert meeting are represented in Table 6.2. The minutes of the experts meeting can be found in Appendix G. In Table 6.2; the red highlighted text indicates that the experts did not accept the suggested Scrum practices for the element, and the green highlighted text indicates that the experts accept the suggested scrum practices for the element. The overall analysis of the expert meeting is done following table 6.2.

Table 6.2 Results of the expert meeting

Scrum practices as a suggestion to the selected complexity elements and problems	**Experts views/opinion on the applicability of the suggestions for HW development projects in the company**
Sprint, Scrum board, Burndown charts High number of project goals Dependency on external stakeholders Uncertainties in scope	Two of the three experts (E2 &E3) agree that suggested Scrum practice can help to manage all complexity elements. However, the expert (E1) does not agree that dependency on external stakeholders can be managed with the suggested Scrum practices as the practice can only help to know the current scenario of the project and to manage this complexity; it requires more than that. Overall, the other two interviewees (E2 & E3) strongly believe that the current traditional approach can also manage the respective complexity elements.
Product backlog, Sprints High number of tasks Level of competition	All of the experts agree that using the suggested Scrum practices can manage the high number of tasks as it helps to determine and visualize all the tasks. However, for the level of competition, all of the experts disagree with the suggestion. According to the experts, the customer plays a major role in managing the complexity element (level of competition). It requires good decision-making skills and customer involvement to manage the complexity element. However, all the experts agree that using these Scrum practices can help giving clarity in defining tasks and visualize where the problem is.
Scrum board, Daily stand-ups Dependencies between tasks Involvement of different technical disciplines Interfaces between different disciplines	According to the experts, all the complexity elements can be dealt with by the suggested Scrum practices. The scum practice will help with a clear picture and continuously align with others. Also, it aids in taking the right decisions and provides correct directions for the team to work. However, according to E1 and E3, team size plays a major in the feasibility of applying the suggested Scrum practices. Therefore, they suggest dividing the large team into separate small teams and then use daily stand-ups.
Product backlog Insufficient requirements unclear requirements	Two of the three experts (E2 & E3) disagree with this suggestion. According to the experts, the product backlog is only feasible when the requirements are clear, as only clear requirements can be translated into the product backlog. Also, experts (E1 and E2) remark that customer involvement is very crucial to manage this problem.
Sprints, Product backlog, Sprint retrospectives High project schedule drive Changing requirements	Two of the three experts (E1 & E2) agree with suggested Scrum practice for managing high schedule drive but remark that the scope of the project is required to be defined. However, the expert (E3) disagrees with this, as there is still a possibility of over-committing to satisfy customers, which affects the time. Also, the experts (E1 & E2) disagree that changing requirements can be managed with the suggested Scrum practices for HW development as the Scrum practice can only help in assessing the changes but not control the changes. Also, for HW development at a certain phase, all requirements have to be fixed, so the suggested Scrum practice will not fix the requirement.
Sprint reviews/retrospectives Strict quality requirements	The experts (E1&E3) agree that using the suggested Scrum practices will help in managing with the strict quality requirements. According to the experts (E1), picking of tasks will give better insights and clarity to the members about the role and project status. However, for the expert E2, picking of tasks does not make a difference as the members are already very responsible. However, for the expert (E2), this suggested scum

	practice would not work, as there are some members/tasks that can always go wrong. In that case, retrospective meetings will also not help with a lack of skills, and picking of tasks does not make a difference.
Daily stand-ups virtual/physical Number of locations Involvement of different time zones Number of different nationalities	All the experts strongly agree that daily stand-ups will surely help solve all the respective complexity elements. All experts agree that this Scrum practice will help solve any issues related to communication and interpretations. However, the expert (E1) disagrees that the daily stand-ups are a way to manage different nationalities in a project. According to the expert, the need for cultural training to understand the cultural difference can only help to manage this complexity.
Defined process/way of working according to the project Incompatibility between different project management methods	Two of the three experts (E1 & E3) disagree that having a defined way of working for the project can manage the particular complexity element. According to the experts, the current management approach used in the company is already flexible and can be adapted to the needs of the project. However, according to the expert (E1 & E3), only by understanding different methods and with frequent communication can help in managing the incompatibility between different methods.

Based on the results from the expert meeting, it is seen that experts are not in favour of the suggested scrum practices for seven out of seventeen elements, as they are not convinced that it will help. A common remark given by the experts is that the traditional project management approach can still manage the mentioned complexity elements/problems. If the experts do not see an advantage of the Scrum method over the current management approach; this makes them not positive about applying the Scrum method for HW development projects. The expert's opinion also shows that the value of implementation for scrum is less for them as they are not impressed with the limited value that scrum brings. However, the experts do have a positive opinion on the applicability of Scrum boards, Scrum tools (burndown-charts), daily stand-ups, retrospectives, and sprints.

6.3 Conclusion of experts meeting

To understand whether the suggested Scrum practices will work practically in the company to manage the respective complexity element and the problem identified, a meeting was set up with three experts individually. Based on the results of the experts meeting, the following main conclusions are drawn:

- *Sprints, Scrum Board, Burndown Charts,* can be applied to manage "uncertainties in scope" and "high number of project goals." For "dependency on external stakeholders," the suggested Scrum practice can be applied, but it can only aid in seeing the current scenario and not fully manage this complexity.
- *Sprints & Product backlog* can be applied to manage "high number of tasks," as it will help to determine and visualise all the tasks. However, for "Level of competition," the suggested Scrum practice is not enough and requires good customer involvement and decision making skills.
- *Scrum board, Daily stand-ups* can be applied for "involvement of different technical disciplines," "dependencies between tasks," and "interfaces between different disciplines."

However, the team size should be considered for the feasibility of the suggested Scrum practices in the company.

- *Product backlog* cannot be applied for “insufficient and unclear requirements,” as clear requirements are crucial for the functioning of the product backlog; moreover, it requires high customer involvement to manage this.
- *Sprints, Product backlog, Sprint retrospectives* can be used to manage “high project schedule drive” provided the scope of the project be fixed. However, there is still a possibility to over-commit, so the applied Scrum practice is not enough. In the case of “changing requirements,” this applied practice is not enough as in HW development, all the requirements have to be fixed in the project at some point. Besides, the suggested Scrum practice can only aid in accessing the changes in requirements and not control or manage it.
- *Sprint reviews/retrospective meetings* can work and be applied to manage *“*strict quality requirements,” as it will give clarity. However, the team member picking their tasks does not mean the member will become more responsible as they are responsible enough already, but it will give clear insights about roles and project status. Also, in case of a lack of skills in the team, even the suggested Scrum practice cannot manage the respective complexity.
- *Daily stand-ups virtual/physical* can be applied to manage the complexity elements, “number of different nationalities,” “involvement of different time zones,” and “number of locations.” However, to fully manage “number of different nationalities,” it also requires cultural training to understand the cultural difference.
- *Defined process/way of working according to the project* cannot be applied to manage the complexity element, “incompatibility between different project management methods.” However, to manage this complexity element requires a clear understanding of the different approaches and frequent communications.

It should be noted that though the experts agree with some of the Scrum practices, they are not very much in favour of implementing the Scrum. The experts wonder why and how it would be better than their current way of working. The expert's remark shows that though they do not use scrum, they are familiar with the agile concept. However, the remarks also show that they are not fully open to implementing scrum method, which will be further discussed in chapter 7.

7 Discussion

This chapter aims to reflect and discuss the findings of the research results. Section 7.1 comprises of the discussion on the findings. Section 7.2 discusses the implication of the findings and their implementation for the company. Finally, section 7.3 discusses the limitations of the research.

7.1 Discussion on findings

The main objective of the research is to find if the Scrum method can fit for HW development projects in the company. The findings are gathered from the literature study and case study at the company. The project management approach, as described by the literature, shows that the traditional project management approach is effective for projects with a predictive and stable environment. However, the traditional approach faces problems because of its formal approach and lack of flexibility. Scrum method, on the other hand, proves to be effective and addresses the solution for an uncertain and volatile environment, which is very much the case for the existing HW development projects. The data gathered from the interviews and complexity assessments illustrate that large HW development project follows a traditional project management approach irrespective of their complexity. Also, it shows that the differences and number of complexity elements did not play a role in choosing the management approach. The scrum method used for SW development are analysed and are seen as a valuable addition to the projects, and the benefits of agile project management approach are in line with the literature studied. As the literature shows the possibility to use the scrum method for HW development, a hybrid approach could be considered. As the scrum method does have benefits, but the differences in HW development compared to SW development makes it difficult to follow the scrum method fully.

Bosch-Rekveldt (2011) developed a general TOE framework for the complexity assessment in the process industry. The TOE framework is qualitative and can be used to assess the type of complexities that can be expected in the HW development project. The project managers generally have a good understanding of the technical challenges, however, since the selected cases belong to the high tech sector, which could be the reason for remarkably different and high complexity score in the complexity assessment.

The results of the interviews show that the company is well versed in planning the HW development with the traditional (waterfall) approach. However, the analysis of cases also shows some common problems currently faced in the management approach of HW development projects (refer to table 5.3). Most of the interviewees also mention that complexity elements that are marked high are not managed well. This shows that the current approach requires some more support to make it effective for HW development projects. The suggested Scrum practices are not fundamentally different from the already applied practices of the traditional approach; however, it adds value.

The expert's remark on the suggestion shows that the Scrum practices (Scrum boards, Scrum tools (burndown-charts), daily stand-ups, retrospectives, and sprints) can be applied in HW development projects. However, the overall remark is that the Scrum practices are and not the solution for some complexity elements. The expert's remark also says that they do not see the advantage of using Scrum practices over the current applied practices, as they are not impressed with the values/advantages scrum method can bring. This could also be influenced by resistance to change. Since the experts are responsible for the management in the company, bringing change in

management when they do not an advantage is not possible. Therefore, to reduce resistance to change in general requires a sense of urgency and a change in the mind-set.

The environment of the organization affects the implementation of agile project management, as there is a significant difference in the way organizations works when following the traditional project management approach. Organizations need to accept the changes created by the implementation of the agile approach and see the value it brings.

Also, the focus of this research was directed towards the scrum method, which is a wise choice. As the SW development team was already using the scrum method, it would be easier to adopt compared to other methods. However, there is still an opportunity to look for other agile practices which could have also bought advantages.

7.2 Implications for the company

With this research, a first step is made towards the possible application of the Scrum method for HW development projects.

The case study clearly illustrates the distinction between traditional and agile project management approaches. The three different projects show noticeable differences in terms of the number of complexities, project characteristics, and problems encountered. The cross-case analysis illustrates that the management approach followed currently for selected projects faces some problems. Overall, the customer has a strong impact on the management approach followed. Besides, the analysis shows that problems also exist in the Scrum method applied for SW development. The problems realized when following Scrum imply that coaching and a shared belief of agile among the members are still lacking and need attention.

Based on the experts meeting, certain practices have proven more effective and, therefore, more important compared to other practices. The following similarities and differences are seen when comparing the practices to the traditional project management approach at the company.

- *Daily stand-up/ Scrum meetings*- The company already follows weekly meetings and, in some cases, daily meetings. However, it needs to be regulated to achieve the full benefits of the practice. Tasks are assigned to the members based on the discussion in the daily meetings. Scrum meeting aims to increase knowledge sharing transparently with others throughout the project.
- *Informal face-to-face communication*- Already followed in the company currently with a one-room approach.
- *Multidisciplinary team with one goal and Division of roles*- Already exists at the company.
- *Sprint reviews/retrospectives*- In the current approach, only at the end of the project, a lessons learned meeting is conducted. Nevertheless, a retrospective/review meeting after every sprint facilitates the learning process and transfers the knowledge with every sprint.
- *Frequent communication with the customer*- Close collaboration with customers throughout the project already exists at the company. However, showing the progress frequently and not when something is achieved should be more emphasized when following Scrum.

- *Scrum board and Scrum tools (Burndown chart/ Release planning)* - Scrum boards or other Scrum tools (burndown charts) are updated daily to visualize the remaining tasks. This is different from the regular planning tools used in the traditional approach.
- *Sprint planning and selection of work, Time boxed sprints*- Aids in early identification, analysis, and management of requirements. The project team should together define project requirements. Also, new requirements are implemented during the next sprints. However, when applying to HW development, the sprint length and the type of deliverable at the end is modified

Some practices are already followed at the company, and even though the differences are small, the effectiveness and frequency in which it is followed are different. To implement the scrum practices at the company, four main scrum practices Daily stand-up meetings, Time boxed sprints, Product backlog, and Sprint reviews/retrospectives are considered. These four practices are considered because they are the most commonly applied practices of the scrum method. Also, applying these four practices requires a change in the organisation compared to others. An adoption procedure of scrum practices are proposed in the following to make it actionable in the company's way of working as to how it is better and adds value to the current way of working and how can it be implemented after considering the remarks of the experts.

Daily meetings

The purpose of daily meetings is to transfer/distribute information, make quick decisions and discuss progress. It is meant to share updates, information and to solve/address specific issues. It is short, retains more focus, and identifies the problems quickly than weekly meetings. The added value is that it is more up to date and team members show more commitment. It acts as a fast and short feedback loop, which helps to solve emerging issues easily. While weekly meetings have different focus like long term planning, financial discussion, and update with the customers.

Based on the experts meeting, the daily stands ups are effective; however, the team size plays an important role. Therefore, if the project team is not too big, then daily meetings are for the whole team, but if it is a large team, then daily meetings are conducted with every discipline and then weekly central team meetings. These meetings allow the teams to synchronize and discuss the incompatibilities and dependencies.

It can be applied throughout all phases of PGP of the company, as there is a need throughout the project.

Time boxed sprints

The purpose of time-boxed sprints for HW development is to help the team stay focused and organised. It helps to keep the requirements focused. In HW development, the duration of sprints can be longer; however, the advantage is to have an agreement of activities, and the team is focused on these activities during each sprint. The added value is that activities are not changed during the sprints, which is makes the scope fixed.

Based on the experts meeting, the scope of the project needs to be certain in the overall project for the implementation of sprints. Also, in the interviews, it was mentioned that it is difficult for the members to focus for a long time; following sprints can bring the focus for a short cycle.

The advice to plan the sprint for HW development is to look at the time of each phase of PGP and see if it is possible to define intermediate deliverables. The project leader and team should look together in each phase if there can be any intermediate deliverables made during one phase of PGP. The intermediate deliverable could be 3D models, 2D designs, completion of design reviews/design document or test plans. It is advisable and also mentioned by one of the interviewees (SW2) to have the sprint length longer in the design phase as it is time-consuming than in later stages of the project.

Product backlog

The purpose of the product backlog for HW development is to manage the changes and change requests coming in during the project. It can be used to specify the order of the deliverables that have to be fulfilled. The added value is that it gives clarity and stability on what to implement when in the project. It aids in managing the scope and changes.

Based on the experts meeting, the product backlog is feasible only when the requirements are clear. So all the request for new requirements that comes in should be first checked by the project leaders about its clarity. Later it should be put in a database that is shared commonly with all the team members, so everyone is aware of what needs to be done. Items from the backlog should be later considered for planning in the integral schedule of HW development from the definition phase.

Sprint reviews/retrospectives

The purpose is to investigate the way of working and inspect what has happened. They are an evaluation meeting during the project, which increases knowledge and learnings to bring improvements. The added value is that learnings from the meeting can be executed earlier and has an effect on the deliverables. It is like a fast feedback loop that can be implemented during the project.

It should be planned and conducted with all the team members, as different perspectives are necessary for finding where the improvements are needed. It can replace lessons learned meetings at the end. Sprint reviews can be conducted during each phase of PGP.

The conclusions drawn based on the expert meeting are an indication that Scrum can fit for HW development projects. Still, a sense of urgency and cultural change is needed to apply Scrum practices in the company. This will take time. The change in management can only start if the company is convinced that suggested Scrum practices would help them in managing the specific complexity elements and problems. However, even after implementing scrum, the involvement of the customer is very crucial for the success of using the Scrum method in development, be it in HW or SW, which is also mentioned by experts. Therefore, the company has to take constant steps to take frequent feedback from the customer and make agreements about the customer's involvement in the early stages of the project.

7.3 Research limitations

There are certain limitations of the research due to the chosen research method (case study) and focused short time of this thesis. The limitations of this research are as follows:

- The practices of Scrum considered in this research are not all of the practices of Scrum but only the most commonly applied practices, and also, there exist other methods of agile. However, due to the time constraints of the research, the composed and limited practices of Scrum for HW development was the right and quick method to review the current management approach and find out if Scrum practices can fit for HW development projects.
- The research was conducted from the company's perspective as the interviewees for the case study, and members of the expert meeting were from the company. Moreover, since only three case studies were conducted, the generalizability of the research results is limited.
- The Scrum practices are suggested based on the complexity elements and problems encountered. The problems are gathered from the semi-structured interviews. However, this leaves room for subjectivity.
- One of the case selection criteria was to select projects that are critical to manage, which show bias risk.
- There exist 47 complexity elements in the complexity assessment, but only those elements that are marked high by two or more interviewers are considered, due to the time constraint of the research. This is a limitation as there might exist other complexity elements in the HW development project, which are high but not experienced by all the interviewees in a similar way, which could/could not be managed by Scrum practices.
- The benefits of Scrum over current applied practices are not measurable, like quantifying the increased efficiency by hours. Therefore, it is difficult to analyze to what extend Scrum practices can help, as there is no quantitative data.
  The research is about suggesting Scrum practices for HW development projects. However, it does not focus on how to implement the suggested Scrum practices in the company. This is considered a limitation, as there might exist some factors that might influence the implementation of the suggested Scrum practice.

8 Conclusion and Recommendation

This is the last chapter of the report. Section 8.1 elaborates on the answers to all research questions and gives the conclusion to the research. Section 8.2 comprises a set of recommendations for the practical implementation and future research. Finally, section 8.3 discusses the reflection on the research process.

8.1 Conclusion

To conclude the research, the main research question has to be answered in the end. The answer to the main research question is combined from the answers of all the sub research questions. Therefore, the sub-research questions are answered one by one.

SRQ1. What are HW development projects?

HW development projects belong to the high tech sector where the end deliverable is an HW product (tangible) developed to the needs of the customer. The products are complex, and it also consists of SW development which acts as a medium between HW and the customer. HW and SW development have similar sub-phases of development, and both have functional and non-functional requirements. There still exists much difference between both, which makes HW development different and more complex than SW development (refer to Table 3.1). Since HW and SW coexist in HW development projects, this creates challenges due to increased interfaces in the project.

The case study gave several other pointers on the characteristics of HW development projects. A wide variety of expertise is required to develop the products; therefore, the project consists of multidisciplinary teams. The time to market and the need to meet the performance requirements (quality) are the main two drivers in the project, and the cost is generally not the focus from the company's perspective. Generally, the HW development projects are managed with the traditional project management approach.

SRQ2. What is the Traditional project management approach used for hardware development projects?

The project organization and management play a crucial role in HW development. Project characteristics are the basis for selecting the right management approach. In this research, the traditional project management approach is defined as a structured management approach with a set of rules and guidelines, tailored to fit the project. The traditional project management approach is suitable for predictable projects with simple and clear boundaries. The planning of the development process, according to the traditional approach, advances linearly. It is sequential, and each phase starts when the previous phase is completed (waterfall model). An improved version of the waterfall model is the stage-gate model consisting of decision points before starting the new phase. The traditional project management approach applied for HW development projects faces difficulties. The coexisting HW and SW make the project very complex, and the trade-off decisions for the successful management of the project are harder to make.

SRQ3. What is the Scrum method, and what are the practices of Scrum that can be applied for hardware development projects?

The Scrum method is a framework of agile project management, which is based on the agile values and principles. The Scrum method was initially developed for SW development and produces working products incrementally for every short duration of time called a sprint. The literature study conducted on the Scrum method indicates that Scrum is suitable and very helpful in case of high uncertainty and changing requirements. It is a framework for product development, made up of various roles, ceremonies, and techniques, which are referred to as Scrum practices.

Based on the agile value and principle, the Scrum practice is related to (refer section 3.5.1), and based on the HW characteristics, the applicability of the Scrum practice for HW development is decided. The theoretical framework of this research summarises the list of practices and benefits that can be applied to HW development projects. Table 8.1 represents Scrum practices and its adaption HW development.

Table 8.1 Scrum practices for HW development

Scrum element	**Can be applied /should be adapted for HW development**
A shared belief in agile within the team	Applied directly HW development has an increased number of interactions and interfaces, which increases the complexity of communication within the teams. The suggested Scrum will act to manage and reduce this complexity.
Coaching of agile	
Daily stand-up/ Scrum meetings	
Informal face to face communication	Applied directly It is the most powerful way of communication, but in the case of distributed teams, it is not possible to follow this practice always, and the involvement of whole team members is crucial to get everyone's opinions. Therefore, it is suggested to have pre-work meetings aiming to solve as much as issues possible before the actual planning starts and make the actual meeting short and focused.
Multidisciplinary team with one goal and Division of roles	Adapted To follow this practice, creating the right team that is empowered to innovate is what is needed. A correct set of members of the team will be able to create new ideas and approaches to solve a problem. The team should combine have what it takes to solve a problem as this will help fix the core cause of the problem.
Sprint reviews/retrospectives	Applied directly Since in HW development, the focus is more on the qualitative improvements, so the need for testing at the end phase can be avoided by continuous review of the delivery.
Frequent communication with the customer	Adapted The customer is considered as an important stakeholder who benefits from the end deliverable. Therefore, early feedback is needed; however, active involvement of the customer is crucial.

One defined process/ Scrum way of working	Adapted HW has critical physical requirements, constraints, and requires more upfront work. Therefore, creating a working deliverable is difficult. The goal is to focus on the output that can be discussed, reviewed, and improved. The maturity of the output marks the progress. In this case, documents are given higher preference.
Sprint planning and selection of work	Adapted Allowing late changes is not possible as HW I less malleable. Therefore, it should be determined which requirements are stable and which are more prone to changes to increase the flexibility to allow changes in the development
Prioritised Backlog	Adapted HW lacks modularity as it is difficult to break into small tasks; therefore, the focus could be more on the big task at a time. The output should be considered as something that could be discussed at the end of the sprint, and the duration of the development cycle should depend on the expected outcome and decided at the start of the cycle.
Time boxed sprints	Adapted HW development requires more lead-time and has a high external dependency; therefore, time constraints of sprints may not work all the time. The strategy is to schedule the overall workflow considering the lead-time and different scenarios.
Scrum board Scrum tools (Burndown chart/ Release planning)	Adapted The output should be considered as something that could be discussed at the end of the sprint, and the duration of the development cycle should depend on the expected outcome and decided at the start of the cycle.

SRQ4. How is the traditional project management applied for HW development projects in practice?

The main purpose of this research sub-question is to get an overview of the management of HW development projects in practice. The PGP of the company is central in the development process for products followed in the build to spec projects. The execution of the projects is based on the PGP that provides a structure to the projects. The PGP of the company is flexible enough to adapt to the needs of the project. The PGP is similar to the stage-gate model of the traditional project management approach, as studied in the literature study (refer section 3.2). The PGP makes it clear to identify the phase and deliverables for each phase in the project. The case study analysis shows that the management approach was not dependent on the project characteristics or complexities, nor did it change due to any problems encountered in the project. The case study analysis captures the following list of management practices that are currently applied for HW development projects in practice:

- The development process, project phase and deliverables for each phase are managed through the PGP of the company

- The project goals are decided together with customer and team
- The planning is done using the work break down structure and project milestones
- Project manager defines the tasks with the key members and assigns the tasks to the members in advance based on the competencies
- The collaboration with the customer starts with the definition of requirements
- Weekly team meetings to maintain frequent communications between all the members
- Requirements changes in HW development is allowed until the design phase after that it is fixed
- Team center is used in the project to share the documents within the team and is an integral part of the development process
- One room approach is adopted in the project so that teams can interact and engage easily
- The project manager manages the interface between HW and SW development with the integration schedule
- At the end of the project, a lesson learned meeting is held with the team and the customer

SRQ5. What practices of Scrum can be adopted for the benefit of HW development projects?

Several steps have been taken to answer this question. All the selected cases are compared with each other to find a list of complexity elements and problems is made from the complexity assessment, project characteristics, and problems identified from the cases (refer to table 5.4). Then the most appropriate Scrum practices were suggested to manage the selected complexity elements and problems based on the theoretical framework. Further, based on the conclusion from experts meeting, the scrum practices that are considered for adoption in HW development projects are *Sprints, Scrum Board, Burndown Charts*, *Daily stand-ups,* and *Sprint reviews/retrospective.* Using these practices is beneficial for the HW development project because it can manage the following complexity elements and problems.

- *Uncertainties in scope* - using the Sprint, Scrum board, and Burndown charts can help to see the progress; total time elapsed and track scope fulfilment. This means that if there is a change in scope, it will be visible to the team. With every new scope change during the sprint, it should be communicated within the team through daily stand-ups, discuss its implications, and plan the future sprints accordingly.
- *High number of project goals*- using sprints helps in achieving all the project goals as it specifies target and helps the team to work towards it. When there are a high number of goals, it is advisable to have scrum boards to visualize and keep track of all the goals and development.
- *High number of tasks* - Creating a product backlog will aid, with a large number of tasks, the suggestion would be feasible to follow one big task at a time of sprint that is considered achievable.
- *Dependencies between tasks, Involvement of different technical disciplines*, *Interfaces between different disciplines* - To manage the dependencies, it is important to know all the dependencies. Using scrum board or other scrum tools like a burndown chart will help visualize the dependencies, track the project status, and specify the interfaces. Also, doing daily stand-ups/scrum meetings is important and suggested, as it will further help team

members to communicate clearly about the dependencies with each other, which will improve the correlation between tasks.

- *Strict quality requirements*- Only with full team effort, the quality will be achieved. Agile practices are developed to achieve high quality and value in the project. Conducting sprint reviews/ retrospective meetings with the team will help to analyze the work done and improve quality. When using sprint planning meetings, which are like the kick-off for starting the sprints, where the whole team meets and together will be able to come up with solutions that help in solving challenges. Therefore, it is important to have the right team.
- *Number of locations, Involvement of different time zones, Number of different nationalities*- All the three complexity elements can have a possible effect on communication and coordination within team members. In this case, it is advisable to use daily stand-ups (virtually/physically) to maintain close collaboration and resolve language barriers and issues with all members.

MRQ. How can the traditional project management approach of HW development projects be improved by using the scrum method?

The answer to the main research question is formulated by combining the answers from all the sub research questions:

Both the traditional project management approach and the scrum method are significant and beneficial. The main aim of blending both approaches is because elements from both approaches achieve better results than other practices. The traditional project management approach provides a defined development process with thoroughness, but the scrum method formalizes more on the execution of the project and strives to produce a better quality of results. The literature study shows that scrum can fit (some can be applied directly, and some need to be adapted) HW development projects. However, the selection of practices should be based on project complexity and characteristics.

In practice, the traditional project management approach used for the HW development project is well established, and the customer has a strong impact on the way of working. The case studies illustrate the distinction between management approaches. The three different projects show noticeable differences in terms of the number of complexities, project characteristics, and problems faced. The cross-case analysis illustrates the main complexity elements and problems encountered in the traditional management approach in the HW development projects. In conclusion, research determines suggestions of the scrum practices and adoption procedure (refer to section 7.2) that can be applied together with the traditional project management approach for HW development. It contributes to managing the specific complexity elements and problems.

8.2 Recommendation

Section 8.2.1 describes the recommendation made for practice. It is the general recommendation given for the project teams of HW development projects, followed by recommendations for future research in section 8.2.2.

8.2.1 Recommendation for implementation of this research in practice

- It is recommended to discuss with the project team about the meaning of agile and practices of Scrum. This is done to make sure that all of the members involved in the project have the same mind-set. This is a way forward to avoid problems related to commitment and teamwork in the project.
- The results of the complexity assessment could be considered as a starting point to determine certain project characteristics. This can further be used to select the right competencies based on the project characteristics. Also, the suggested scrum practices should not be followed fully by the book but should be adjusted more according to the needs and demands of the project.
- When applying the scrum for HW development, not many members have enough knowledge to do it. However, inputs from experts of the scrum for HW development should be considered in the pilot project for better understanding of the concept and workload.
- Based on the suggested scrum practices for HW development, a full implementation plan should be made depending upon the needs of the project and should be given to all the members. When following a pilot project with this plan, the practicality of the scrum practices can be tested and modified.
- Changes are inevitable in complex projects. Changes should be embraced rather than being avoided or restricted. Using agile practices will give flexibility and increase satisfaction for the customers.

8.2.2 Recommendation for the future

Since not much literature is available exactly specifying practices that can be adopted for HW development, this study can be used as the basis for future research.

- The study suggests fitting Scrum practices to HW development projects. To further test the results, it can be applied directly on a pilot project and compared with another project where a traditional approach is followed. The lessons learned from both the projects should be compared with each other.
- It is to be noted that there are frameworks of agile other than the Scrum method. It is recommended to compare the benefits of the Scrum method with other frameworks of agile and check, which is more effective and compatible with HW development projects.
- The term agile is common yet uncommon. That is to say, because most of the interviewees knew about the word agile because of the popularity of agile/Scrum yet were not fully aware of the concept/process and how it is implemented. Therefore, it is recommended to check the agility of the project based on the knowledge and nature of the members.
- Since customer involvement is very crucial for the success of the scrum method, therefore, it would be interesting to learn more about the strategies of customer engagement.
- One common observation of factors that affects the implementation of Scrum method/practices is the resistance to change. Therefore, future research could be on finding reasons for resistance to change when it comes to agile and formulating suggestions to solve it. This would help in the applicability of agile project management approach in practice.

8.3 Reflection

Scrum for hardware development projects, a challenge I took for my thesis. With not a lot of knowledge of hardware development projects, I am very grateful to the company for giving me this opportunity to learn and grow.

I had projected to start my thesis by November 2019, and it fell into place. The proposal phase was challenging since it was a completely new topic. The first couple of months were very crucial because the interview questions had to be designed and carried out. The onset of the pandemic put things to a halt a bit; however, I was fortunate that the interviews were completed before the intelligent lockdown. Only the analysis and expert meeting were to be completed. Theoretically, working from home should have been easy.
Nevertheless, working from home was a challenge and demotivating. Having such little contact with people and seeing the conditions getting worsen caused stress and anxiety. This caused much delay in my time planning.

Eventually, I realized there was not a lot I could do. With the help and motivation from my supervisors, I have managed to reach here today. The underestimation of writing the thesis report was an error on my part. I now understand that writing the report is an iterative work. In regards to my writing, the lack of clarity and complexities of the sentences was the main feedback. Though it sounded perfect in my head, putting my thoughts together on paper was difficult for me.

Each phase did not go as per the plan, but I am pleased with my learnings. While working, I understood the interdisciplinary aspects of projects and the importance of coordination. It also helped me in developing a specific skill set, like how a company works or how to interact with higher officials, which I think would be helpful to me in the future as I do not have prior work experience.

References

Albers, A., Behrendt, M., Klingler, S., Reiß, N., & Bursac, N. (2017). Agile product engineering through continuous validation in PGE–Product Generation Engineering. Design science, 3.

Antoniadis, D. N., Edum-Fotwe, F. T., & Thorpe, A. (2011). Socio-organo complexity and project performance. *International Journal of Project Management*, *29*(7), 808-816.

Atzberger, A., & Paetzold, K. (2019, July). Current challenges of agile hardware development: What are still the pain points nowadays?. In *Proceedings of the Design Society: International Conference on Engineering Design* (Vol. 1, No. 1, pp. 2209-2218). Cambridge University Press.

Augustin, L., & Schabacker, M. (2019, July). Combining Scrum and Design Thinking for a Highly Iterative and User-Centric Hardware Development Project. In *Proceedings of the Design Society: International Conference on Engineering Design* (Vol. 1, No. 1, pp. 2189-2198). Cambridge University Press.

Baccarini, D. (1996). The concept of project complexity—a review. *International journal of project management*, *14*(4), 201-204.

Bianchi, M., Marzi, G., & Guerini, M. (2020). Agile, Stage-Gate and their combination: Exploring how they relate to performance in software development. *Journal of Business Research*, *110*, 538-553.

Bosch-Rekveldt, M. G. C. (2011). Managing project complexity: A study into adapting early project phases to improve project performance in large engineering projects.

Brandl, F. J., Kagerer, M., & Reinhart, G. (2018). A Hybrid Innovation Management Framework for Manufacturing–Enablers for more Agility in Plants. *Procedia CIRP*, *72*, 1154-1159.

Cho, J. (2008). Issues and Challenges of agile software development with SCRUM. *Issues in Information Systems*, *9*(2), 188-195.

Ciric, D., Lalic, B., Gracanin, D., Palcic, I., & Zivlak, N. (2018, March). Agile project management in new product development and innovation processes: challenges and benefits beyond software domain. In *2018 IEEE International Symposium on Innovation and Entrepreneurship (TEMS-ISIE)* (pp. 1-9). IEEE.

Cohn, M., & Ford, D. (2003). Introducing an agile process to an organization [software development]. *Computer*, *36*(6), 74-78.

Cooper, R. G., & Sommer, A. F. (2016). The Agile–Stage-Gate hybrid model: A promising new approach and a new research opportunity. *Journal of Product Innovation Management*, *33*(5), 513-526.

Charvat, J. (2003). Project Management Methodologies: Selecting, Implementing, and Supporting Methodologies and Processes for Projects. Hoboken, NJ: John Wiley & Sons, Inc.

Cooper, R. G. (2001). Winning at new products: accelerating the process from idea to launch.

Deemer, P., Benefield, G., Larman, C., & Vodde, B. (2012). A lightweight guide to the theory and practice of scrum. *Ver*, *2*, 2012.

Dingsøyr, T., Dybå, T., & Moe, N. B. (Eds.). (2010). *Agile software development: current research and future directions*. Springer Science & Business Media.

Dingsøyr, T., Dybå, T. and Abrahamsson, P. (2008) 'A preliminary roadmap for empirical research on agile software development', *Proceedings - Agile 2008 Conference*, pp. 83–94.

Drechsler, R., & Breiter, A. (2007, July). Hardware Project Management-What we Can Learn from the Software Development Process for Hardware Design?. In *ICSOFT (SE)* (pp. 409-416).

Eloranta, V. P., Koskimies, K., & Mikkonen, T. (2016). Exploring ScrumBut—An empirical study of Scrum anti-patterns. *Information and Software Technology*, *74*, 194-203.)

Fuchs, C. A., & Golehhofen, F. J. (2019). Mastering disruption and innovation in product management. *Switzerland: Springer International Publishing AG*.).

Gary, K., Enquobahrie, A., Ibanez, L., Cheng, P., Yaniv, Z., Cleary, K., Kokoori, S., Muffih, B. and Heidenreich, J. (2011) 'Agile methods for open source safety-critical software', Software: Practice and Experience, Vol. 41, No. 9, pp.945–962.

Garzaniti, N., Briatore, S., Fortin, C., & Golkar, A. (2019, March). Effectiveness of the Scrum Methodology for Agile Development of Space Hardware. In *2019 IEEE Aerospace Conference* (pp. 1-8). IEEE.

Geraldi, J., Rodney Turner, J., Maylor, H., So¨derholm, A., Hobday, M. and Brady, T. (2008), "Innovation in project management: voices of researchers", International Journal ofProject Management, Vol. 26 No. 5, pp. 586-9.Gane, C. (2001). Process Management: Integrating Project

Gloger, B., & Häusling, A. (2011). *Erfolgreich mit Scrum-Einflussfaktor Personalmanagement: Finden und Binden von Mitarbeitern in agilen Unternehmen*. Hanser.

Haas, K.B. (2009), Managing Complex Projects: A New Model, Management Concepts, Vienna, VA.

Hanser, E. (2010). *Agile Prozesse: Von XP über Scrum bis MAP*. Springer-Verlag.

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

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

High-Tech Industry in the Netherlands. (2016, December 5). Retrieved from https://dutchglory.com/high-tech-industry-in-the-netherlands/

Highsmith, J. (2009). *Agile project management: creating innovative products*. Pearson education.

Hornstein, H. A. (2015). The integration of project management and organizational change management is now a necessity. *International Journal of Project Management*, *33*(2), 291-298.

Hoda, R., Noble, J., & Marshall, S. (2008). Agile project management. In *New Zealand Computer Science Research Student Conference, NZCSRC 2008.*

Hendriks (2019) http://resolver.tudelft.nl/uuid:1d1b115b-149b-4718-87b4-ad1c1a400bc6

Jugdev, K., Thomas, J., & Delisle, C. (2001). Rethinking project management–Old truths and new insights. Project Management Association Finland/Norwegian Project Management Forum.

Karlstrom D. & Runeson, P. 2006. Integrating agile software development into stage-gate managed product development. Empirical Software Engineering 11 (2), 203–225.

Lima, G. L. B., Ferreira, G. A. L., Saotome, O., da Cunha, A. M., & Dias, L. A. V. (2015, April). Hardware development: Agile and co-design. In *2015 12th International Conference on Information Technology-New Generations* (pp. 784-787). IEEE.

Livermore, J. A. (2007, March). Factors that impact implementing an agile software development methodology. In *Proceedings 2007 IEEE SoutheastCon* (pp. 82-86). IEEE.

Mahnic, V. (2011). A case study on agile estimating and planning using scrum. *Elektronika ir Elektrotechnika*, *111*(5), 123-128.

Mahnic, V., & Zabkar, N. (2012). Measuring progress of scrum-based software projects. *Elektronika ir elektrotechnika*, *18*(8), 73-76.

Marcal, A. S. C., Soares, F. S. F., Belchior, A. D., & de Freitas, B. C. C. (2007, March). Mapping CMMI project management process areas to SCRUM practices. In *31st IEEE Software Engineering Workshop (SEW 2007)* (pp. 13-22). IEEE.

Maximini, D., Maximini, & Rauscher. (2018). *Scrum Culture*. Springer International Publishing AG, part of Springer Nature.

Morampudi, N. S., & Raj, G. (2013). Evaluating strengths and weaknesses of agile scrum framework using knowledge management. *International Journal of Computer Applications*, *65*(23).

Mushtaq, Z., & Qureshi, M. R. J. (2012). Novel hybrid model: Integrating scrum and XP. *International Journal of Information Technology and Computer Science (IJITCS)*, *4*(6), 39.

Overhage, S., Schlauderer, S., Birkmeier, D., & Miller, J. (2011, January). What makes IT personnel adopt scrum? A framework of drivers and inhibitors to developer acceptance. In *2011 44th Hawaii International Conference on System Sciences* (pp. 1-10). IEEE.

Permana, P. A. G. (2015). Scrum method implementation in a software development project management. *International Journal of Advanced Computer Science and Applications*, *6*(9), 198-204.

Punkka, T. (2012, October). Agile hardware and co-design. In *Embedded Systems Conference* (pp. 1-8).

Paasivaara, M., Durasiewicz, S., & Lassenius, C. (2008, August). Distributed agile development: Using scrum in a large project. In *2008 IEEE International Conference on Global Software Engineering* (pp. 87-95). IEEE.

PMI. (2018). Pulse of the Profession 2018: Success in disruptive times.

Reynisdottir, T. (2013). *Scrum in Mechanical Product Development-Case Study of a Mechanical Product Development Team using Scrum* (Master's thesis).

Royce, W. (1970). Managing the development of large software systems: Concepts and techniques. In: Proc. IEEE WESCOM. IEEE Computer Society Press, Los Alamitos.

Sakal, M. W. (2005). Project alliancing: a relational contracting mechanism for dynamic projects. *Lean Construction Journal*, *2*(1), 67-79.

Serrador, P., & Pinto, J. K. (2015). Does Agile work?—A quantitative analysis of agile project success. *International Journal of Project Management*, *33*(5), 1040-1051.

Schwaber, K., & Sutherland, J. (2017). The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game.(2011). *Available: scrum. org*.

Schwaber, K., & Beedle, M. (2002). *Agile software development with Scrum* (Vol. 1). Upper Saddle River: Prentice Hall.

Schuh, G., Schröder, S., Lau, F., & Wetterney, T. (2016, September). Next generation hardware development: Requirements and configuration options for the organization of procurement activities in the context of Agile new Product Development. In *2016 Portland International Conference on Management of Engineering and Technology (PICMET)* (pp. 2583-2591). IEEE.

Shenhar, A. J. (1993). From low-to high-tech project management. *R&D Management*, *23*(3), 199-214.

Sohi, A. J., Hertogh, M., Bosch-Rekveldt, M., & Blom, R. (2016). Does lean & agile project management help coping with project complexity?. *Procedia-Social and Behavioral Sciences*, *226*, 252-259.

Špundak, M. (2014). Mixed agile/traditional project management methodology–reality or illusion?. *Procedia-Social and Behavioral Sciences*, *119*, 939-948.

Streule, T., Miserini, N., Bartlomé, O., Klippel, M., & De Soto, B. G. (2016). Implementation of scrum in the construction industry. *Procedia engineering*, *164*, 269-276.

Sutherland, J., & Ahmad, N. (2011). How a traditional project manager transforms to Scrum: PMBOK vs. Scrum. In *Agile Congress*. Salt Lake City: Scrum.

Sutherland, J. (2001). Inventing and Reinventing SCRUM in five Companies. *Cutter IT journal*, *14*, 5-11.

Toljaga-Nikolic, D., Petrovic, D., & Mihic, M. (2017, September). How to choose the appropriate project management approach?. In *2017 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT)* (Vol. 2, pp. 1-5). IEEE.

Thornhill, A., Saunders, M., & Lewis, P. (2009). *Research methods for business students*. Prentice Hall: London.

Tomar, A. K. (2017). *Study of scrum framework usage in non-software systems* (Doctoral dissertation, Massachusetts Institute of Technology).

Williams, T. (2005). Assessing and moving on from the dominant project management discourse in the light of project overruns. IEEE Transactions on Engineering Management, 52(4), 497–508

Weinreich, R., Neumann, N., Riedel, R., & Müller, E. (2015, September). Scrum as method for agile project management outside of the product development area. In *IFIP International Conference on Advances in Production Management Systems* (pp. 565-572). Springer, Cham.

Yin, R. K. (2014). *Case study research: Design and methods (applied social research methods)*. Thousand Oaks, CA: Sage publications.

Appendix A

The four key values of agile manifesto are as follows:

1. **Individuals and interactions** over process and tools
2. **Working software** over comprehensive documentation
3. **Customer collaboration** over contract negotiation
4. **Responding to change** over following a plan

The 12 principles formulated are as follows:

1. High priority to **satisfy the customer** through early and continuous delivery.
2. Always welcome **changing requirements** for the competitive advantage of the customer.
3. Deliver **working software** in a frequent and prescribed shorter timeframe.
4. **Business people and developer working together** daily throughout the project.
5. Build projects around **motivated individuals** and trusting them for the job to be done by creating an environment and support for them.
6. **Face to face communication** is considered as the most efficient and effective form of communication for conveying the information.
7. Progress measuring by **working software** as the primary source.
8. Promotion of **sustainable development** by maintaining constant pace indefinitely.
9. Enhance **agility** through continuous attention to technical excellence and good design.
10. **Simplicity** is essential.
11. **Self-organizing teams** aid in bringing best from the architecture, requirements and designs.
12. **Team reflection** on regular intervals to become more effective and then adjusting accordingly.

Appendix B

Relating the scum benefits from table 3.3 to themes in project management

Ability to manage the changing requirements	Requirements management
Detailed estimations for tasks	Task defination
Improved planning and estimation	Project planning
Focus on integration/interfaces in the project	
Creates project plan with shared responsibility	
Less detailed planning, specification and documentation	
Controls scope creep	Scope management
Reduces time and resources	Time & Cost management
Reduces cost and time taken in the project	
Reduces the time required to plan	
Improved communication between the team members	Team & customer engagaement
Achieves customers expectation	
Member of the project have a strong centralised work environment	
Flexibility in the development process	Product developement and evaluation
Improved visibility in project	
Early feedbacks of customer integrated into the development process	
Evaluated work throughout the process	
Reveals deficiency early in the process	Quality
Early testing of physical functional prototypes	
Higher effectiveness to the predevelopment stages	
Improved quality of the deliverable	Goals & delievrables

Appendix C

A. About you and the project (10 mins)

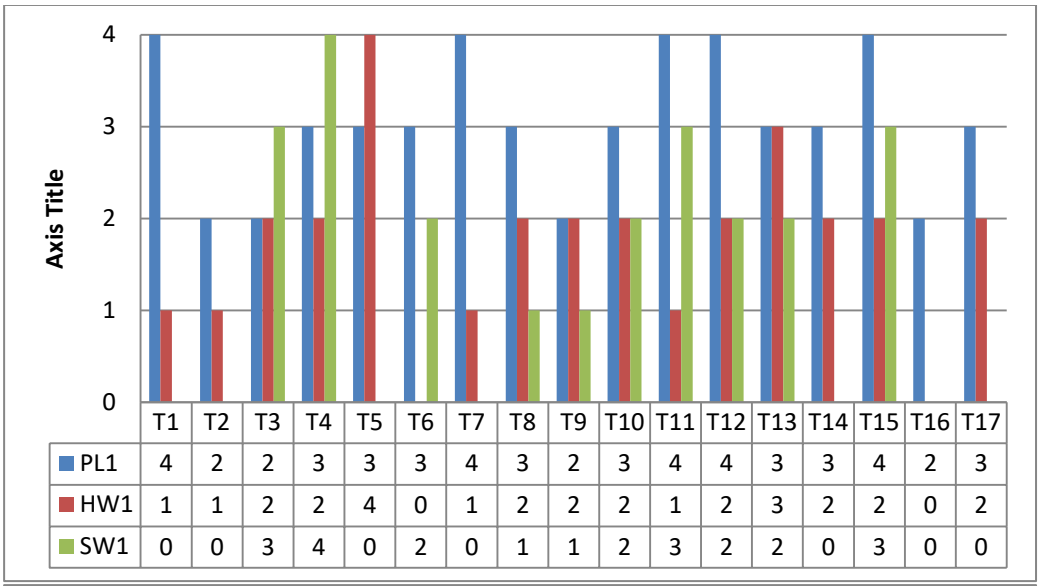
1. What is the project about? (type)
2. What is your role/ agile role/position in the project? How many players are in your team?
3. How many years of experience do you have for this role?
4. Does the project have just SW/ HW development or both?
   4.1. At what stage do the SW and HW integrate into your project?
   4.2. Which part is in the lead?

B. About the Project Management approach (50mins)

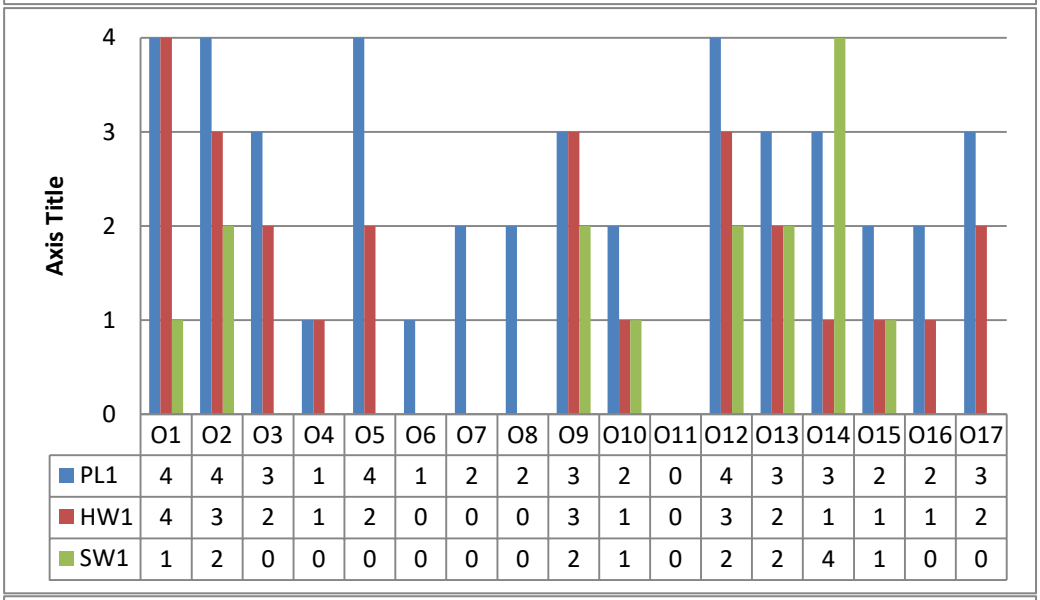
1. Which project management method was chosen for the SW/HW development?
2. Based on what is the project management approach is chosen?
3. How was the project planned?
4. What are the fixed and floating variables in your project in terms of Scope, Time, cost, and quality
5. How were the project scope, time cost, and quality defined and controlled?
6. Which is given priority in terms of Scope, Time, cost, quality?
   6.1. Given the priority, were you able to successfully deliver the product?
7. How were the project goals and deliverables defined?
8. How were the tasks/activities in the project defined and distributed to the team?
9. What do you do if the requirements from the customer are not clear?
   9.1. What was your approach to get requirements clear to the whole team?
10. How were the requirements managed and prioritised?
    10.1. Until what phases were you flexible in the process in terms of changing requirements and expectations from the customer?
11. What were your approach to engage the customer and the team?
    11.1. How do you get the feedback from the customer and team about the progress in development?
12. Did you evaluate in between/after the project? And if yes, then how?
13. What are the advantages of using the approach?
14. What are the disadvantages of using the approach?
    14.1. How could the approach be improved?
15. How is the interface between software and hardware parts of the project managed?
16. What are the challenges of different management approach in SW and HW in one project?
17. What do you know about agile and scrum? (only for the HW dev)
18. What elements of Scrum can be applied to HW dev projects and in which phase and why?

C. How effective was your approach in managing complexities that are marked high?

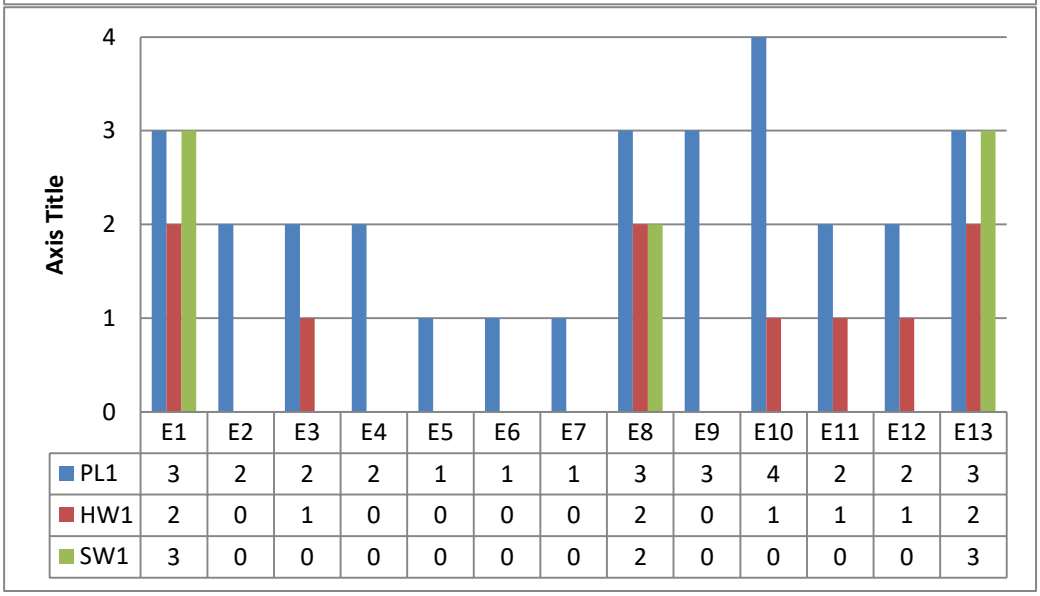
Appendix D



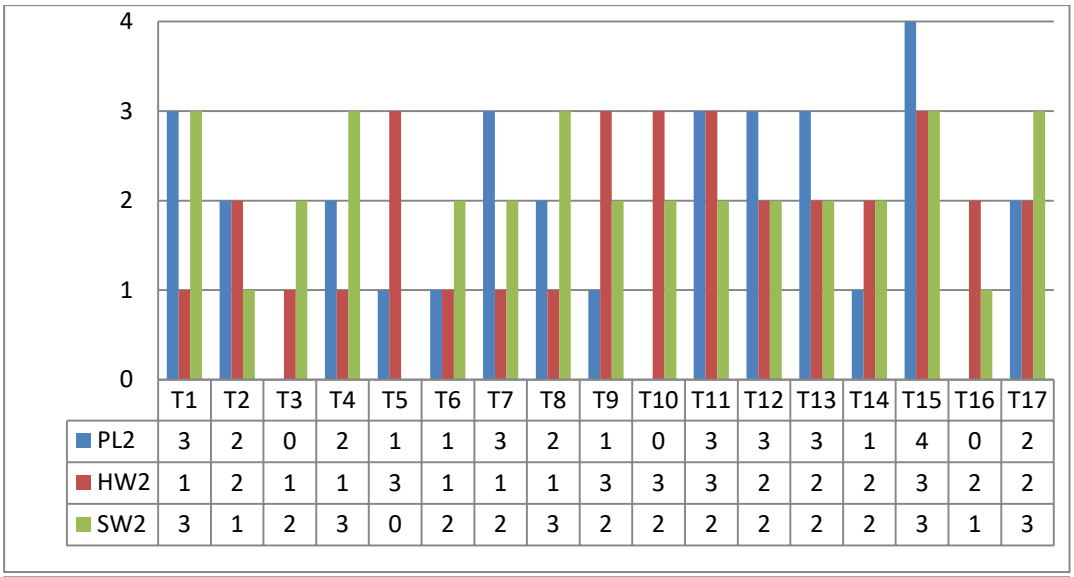
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17
PL1	4	2	2	3	3	3	4	3	2	3	4	4	3	3	4	2	3
HW1	1	1	2	2	4	0	1	2	2	2	1	2	3	2	2	0	2
SW1	0	0	3	4	0	2	0	1	1	2	3	2	2	0	3	0	0



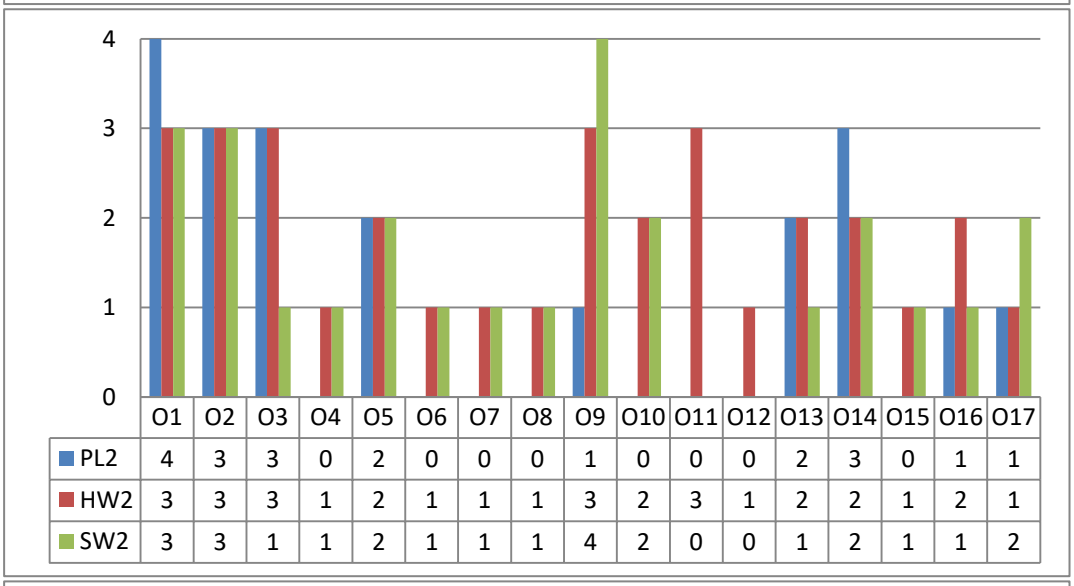
	O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12	O13	O14	O15	O16	O17
PL1	4	4	3	1	4	1	2	2	3	2	0	4	3	3	2	2	3
HW1	4	3	2	1	2	0	0	0	3	1	0	3	2	1	1	1	2
SW1	1	2	0	0	0	0	0	0	2	1	0	2	2	4	1	0	0



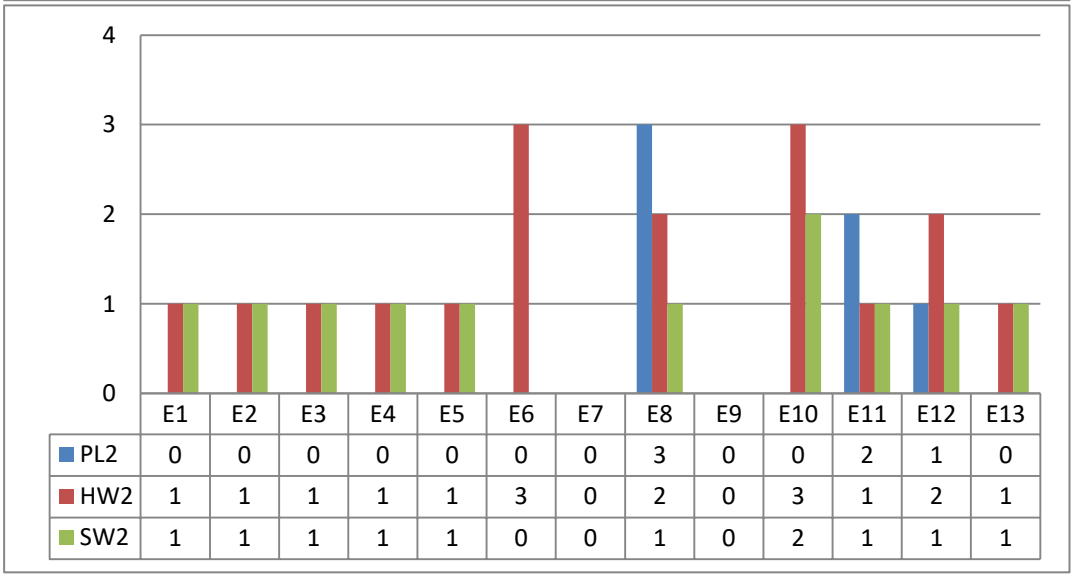
	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13
PL1	3	2	2	2	1	1	1	3	3	4	2	2	3
HW1	2	0	1	0	0	0	0	2	0	1	1	1	2
SW1	3	0	0	0	0	0	0	2	0	0	0	0	3



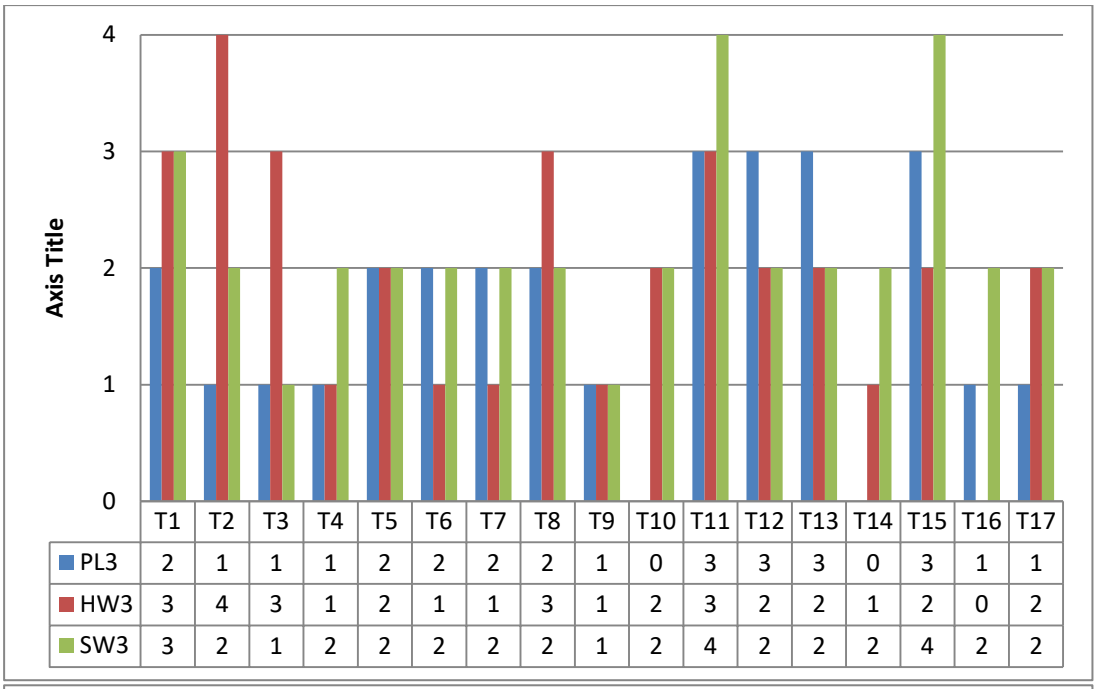
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17
PL2	3	2	0	2	1	1	3	2	1	0	3	3	3	1	4	0	2
HW2	1	2	1	1	3	1	1	1	3	3	3	2	2	2	3	2	2
SW2	3	1	2	3	0	2	2	3	2	2	2	2	2	2	3	1	3



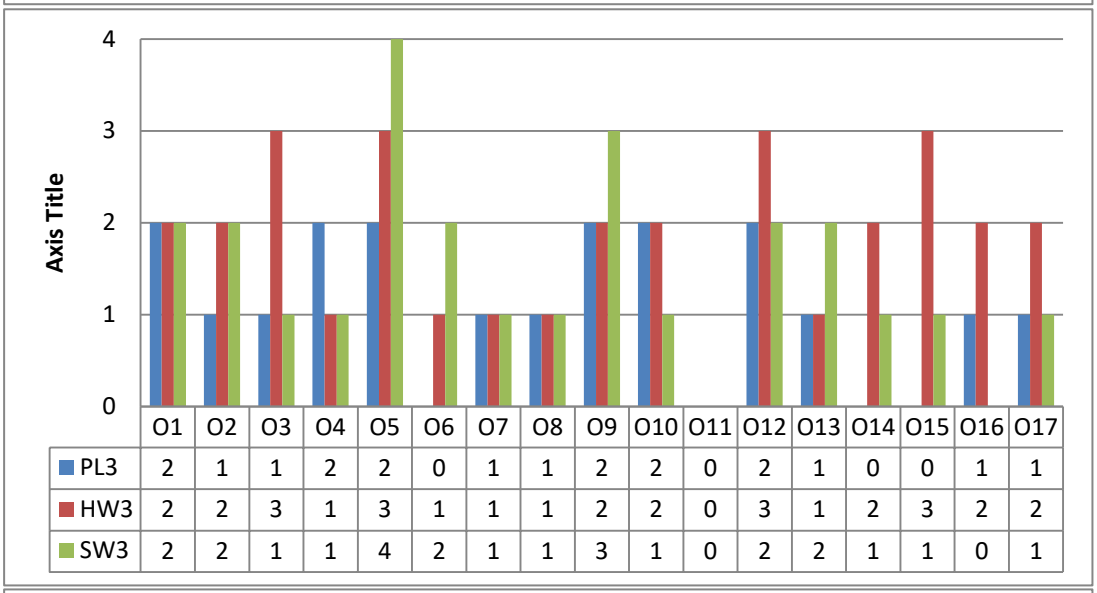
	O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12	O13	O14	O15	O16	O17
PL2	4	3	3	0	2	0	0	0	1	0	0	0	2	3	0	1	1
HW2	3	3	3	1	2	1	1	1	3	2	3	1	2	2	1	2	1
SW2	3	3	1	1	2	1	1	1	4	2	0	0	1	2	1	1	2



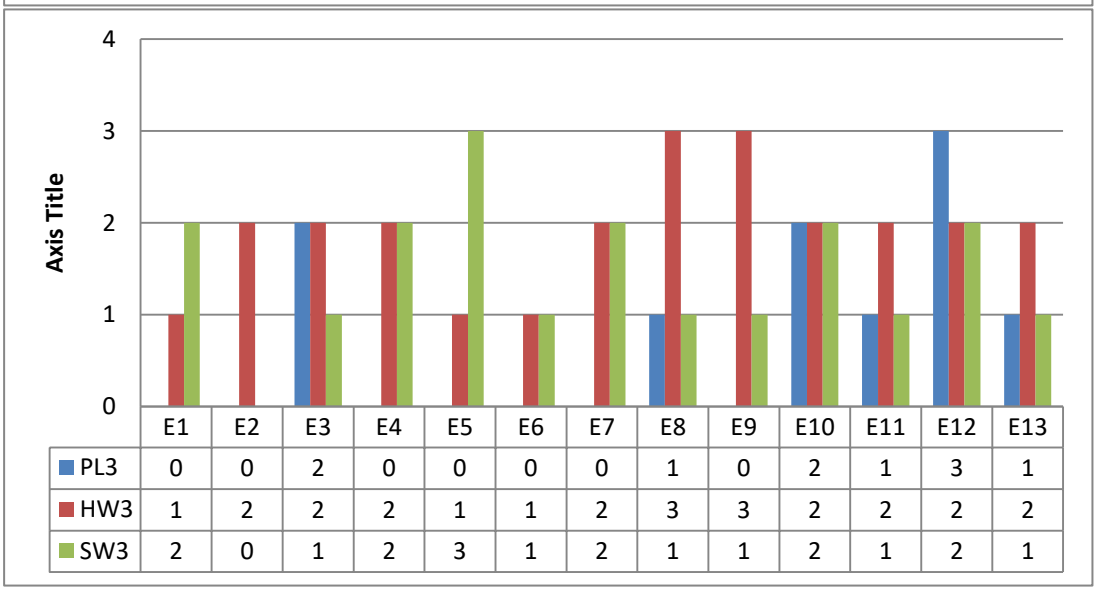
	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13
PL2	0	0	0	0	0	0	0	3	0	0	2	1	0
HW2	1	1	1	1	1	3	0	2	0	3	1	2	1
SW2	1	1	1	1	1	0	0	1	0	2	1	1	1



	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17
PL3	2	1	1	1	2	2	2	2	1	0	3	3	3	0	3	1	1
HW3	3	4	3	1	2	1	1	3	1	2	3	2	2	1	2	0	2
SW3	3	2	1	2	2	2	2	2	1	2	4	2	2	2	4	2	2



	O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12	O13	O14	O15	O16	O17
PL3	2	1	1	2	2	0	1	1	2	2	0	2	1	0	0	1	1
HW3	2	2	3	1	3	1	1	1	2	2	0	3	1	2	3	2	2
SW3	2	2	1	1	4	2	1	1	3	1	0	2	2	1	1	0	1



	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13
PL3	0	0	2	0	0	0	0	1	0	2	1	3	1
HW3	1	2	2	2	1	1	2	3	3	2	2	2	2
SW3	2	0	1	2	3	1	2	1	1	2	1	2	1

Appendix E

Themes		PL1	HW1	SW1
HW/SW development		waterfall approach for HW development and scrum for SW development. HW needs to be thought well before manufacturing and requires a lot of worko validate the design and therefore an iterative way of working will not be suited.	Waterfall approach for HW development more of sequential and also because of the unavailability of designers to work in parallel.	scrum for SW development but not fully (no product owner, evaluation meeting, and sprint review). Also full commitment is required to follow the sprint and not do anything else and this is currently not possible in the company
Project planning		PGP to plan. Estimations based on efforts	Done by PL1, followed the given PGP and PMP	estimates on hours but was underestimated due to unclear requirements, so we made up some by ourselves
Project promises	Time	Highest priority as the customer had a fixed deadline.	Fixed and highest priority but delivered with delays. Though time was given the most priority, quality should have been the priority as the customer, in the end will be fine with delays but not with the product that does not work	Highest priority and defined based on the requirements
	Cost	less relevant and not fixed	Cost is a floating variable	Defined based on the requirements
	Quality	Important but not the highest priority	Not achieved due to the time pressure but later improved with time	Highest priority and defined based on the requirements.
Scope		Not fixed till design phase	Fixed but one interface missed to notice in the beginning then the customer asked to make it.	Inputs are taken from the customer and team to define it. But customer was not involved much to give inputs they did not have the all requirements.
Overall project success	Positive	Overall good	Overall good delivered with required quality	Overall done
	Negative	Time was delayed as customer were not ready on the required time.	Time was delayed	We were not in time because of problems with suppliers and because of our bad approach.
Project goals and deliverables		Defined with the customer	Defined with the customer	Goal is software release. Deliverables are identified by the structure of SW and defined with the team
Tasks		Identified by WBS and distributed based on the discussions with the core team about the right competence.	Task is assigned based on the competence. PL1 defines the priority at higher level. For new tasks in later stages, it is defined in the weekly or daily meetings and assigned through emails.	Defined through stand up and planning sessions up to 3 weeks. From the whole list of work to do team, take their own task based on their competencies. We use Jira, but not really updated.
Requirements		Changes from the customer until the end of the design phase. But prototype projects are expected to get changes. Requirements prioritised by the PL1 based on the critical item.	Managed through the documents, communicated at the start and only key members were involved in defining it. The initial requirements from the customer were not complete therefore; it took a lot of time in discussing. There was one change in requirements from the customer in the detailing phase, but nothing much could be done, as it was the mistake of the customer for	It was not managed; we did not prioritise it, as we did not have everything. We were focusing more on deliverables. I did it with other sys architect and team. Changes were accepted from the customer until the integration phase, which was too late.

			not communicating it before.	
Customer engagement			weekly focus meetings to discuss the progress with the program manager and project leader from the customer side	Weekly meetings, Status update, and tech discussions with customer. My team was not really in contact with the client.	weekly meeting, sometimes daily during the testing phase but sometimes there was a long time of no contact as the customer was not prepared. It is important to get to know the client then no need of f2f. Progress is communicated with the customer through PL1.
Team engagement			Was challenging since it was a big team but one room approach helped	Everyday discussion with my team. Weekly meetings and conference call with the key player of the team, key members of other disciplines discussing the current scenario and where the support from the customer is needed.	Daily and weekly meetings and teams do not know about the progress of other teams, which needs to be improved, as we are dependent on other teams. We use Jira for progress visualization but is not used enough.
Project evaluation			Several internal evaluations and twice evaluations with the customer was conducted for the teams in both locations of the company	evaluations were only done by PL1 for the whole team but no evaluation was done with the HW team separately	Evaluations were only done by PL1 for the whole team but no evaluation was done with the SW team separately which needs to be started.
Project management approach	Advantages		It is less iterative, though more time was taken the quality was better in the end. Also with PGP, milestones can be planned upfront and everyone knows this way of working.	I was also the designer in the team, so easier to manage the small team	with the one-room approach and daily stand up meetings the communication was good
	Disadvantage		it is difficult for the team to focus on the deliverables for a long duration	lack of resources that led to this way of working and this affected the time and the	no clear understanding of what to do. Could be improved
	Improvement		By having faster testing methods	By giving less importance to time and more to quality.	by start working together as one team with MSD (other team) because of dependencies. Visualization with scrum could be improved. Also, teach other teams about scrum.
Interface	definition		Managed by PL1 by making integration schedule with which everyone knows what is available when and also gives a start for further discussions.	Interface managed by the PL1.	Was challenging. Interface could have been better if the communication was maintained well with others from the start.
	Different management approach in one project		Challenge was in aligning the milestones of HW and SW due to different way of working	no interaction with the SW team as there was no dependency	Lack of understanding in the way of working among different teams. Also other disciplines are afraid of micromanagement when using scrum
Scrum			knows about scrum to some extend and thinks daily meetings from could be adapted.	does not know about agile or scrum but is curious in knowing how it can be applied to HW	everything from scrum can be applied to HW but the remark is that everything should be fixed from the start and cannot keep changing in HW unlike SW

Themes		PL2	HW2	SW2
HW/SW development		Scrum way of working was adapted partly for the HW development (all decision were taken in the daily stand up) and fully for the SW development. This approach was chosen to reduce and manage the risks of the project.	For HW development, it is a waterfall way of working as a new phase can start only when the previous phase is fully completed. It is not a choice	Scrum was applied for SW development because it was desired by the company had to be adapted as the customer's way of working was the waterfall approach.
Project planning		PGP was used to see what to do in every phase and decide on the milestones. The tracking of activity was done by Jira, the backlog was created and sprints were planned. outsourcing decisions were based on the available competencies	PGP was used but due to tight schedule the phases were planned simultaneously with assumptions that all will work well	desired delivery dates were gathered and then WBS was created by the members working on the tasks and see if it fits or not within the desired time
Project promises	**Time**	Fixed and highest priority.	Fixed and highest priority.	Overall fixed it is based on time estimates made on the tasks and this was done in Jira
	Cost	effort-based and estimates for the upcoming quarter were given before	most flexible and was effort-based quotes.	not fixed and not given much importance
	Quality	Fixed	Given highest priority but it improved with time	Better quality was expected by the customers in the testing phase but then the customer way of working is to have the product working first and later focus on improving
Scope		Fixed and highest priority. The customer decides on the specifications from that the requirements and scope was decided	Fixed	Fixed
Overall project success	**Positive**	Overall good	Overall good	Overall good
	Negative	Time was escalated.		
Project goals and deliverables		Based on SOW made by the customer, which is reviewed by the management team of the company. Also, PGP has a set of deliverables and milestones	Defined based on the PGP and also followed the customer's way of working which is also similar to PGP	goals are defined in the sprint start meetings by the software team lead with the team based on WBS sent by the team.
Tasks		For every package, WBS is made with all the disciplines involved. They are then divided into small tasks by the competence leaders and	Tasks are defined with key members at the start of the project and assigned based on the competences.	PL2 defines the tasks and decides who is best suited for the task based on the experience.

		then given it to the teams		
Requirements		Each document is developed one after the other, and this process followed from the definition phase to the design phase and is not changed once it is approved.	Defined based on direct communication with the customer. It should be clear from the early stages of the project as it gives clarity about the performance specifications.	The requirements are prioritised based on the wish list of the customer but if something needs to be done first then that's done.
Customer engagement		Weekly progress meeting about each package and issues if any and weekly issue resolution to track the issues	the customer representative was present every week and sometimes daily communication on the phone	The interaction with the customer was once a week by the SW team lead
Team engagement		scrum meetings were conducted with teams per package. Also, project meetings were conducted to give the update about other teams but this was not regular	mostly informal. There was daily scrum meetings with part of the team (SW team and other key people) and also weekly project meetings and discussions through emails	Interaction with the team in daily stand up meetings. The SW team was not part of the progress meetings with the customer unless there is a request
Project evaluation		evaluation meetings were conducted only with the SW team. For others, it was open to discuss whenever needed. meetings with the customer were conducted once at the end.	evaluations were only done by PL2 for the whole team but no evaluation was done with the team separately.	Retrospective meetings are done with the team but they miss the step to implement the improvements.
Project management approach	**Advantage**	helps in focusing on what needs to done and makes everyone work together, discuss about issues, risks and act faster towards it	using PGP makes it clear what is to be done in all phases	can easily go back to the designing phase. Also with daily meetings, it helped in knowing the dependency amongst the team members
	Disadvantage	the team loses track and don't achieve the milestone when focused for a long time in the end. Also in SW teams, incomplete tasks in sprint move to the next sprint without focusing on how to achieve it	very formal as new phase cannot start even if something small is skipped. PGP is controlled by the management team of the company and project is run by customers which causes decoupling	scrum needs to adapt to the project and it is not clear on how to adapt. Sprint length should depend on the work/phase of the project as having the same sprint length in the whole project is not flexible
Interface		Managed by maintaining a lot of documents and necessary communication through emails. The challenge was in matching milestones in both HW and SW because of different ways of working	managed by the PL2 and one-room approach helps in the working effectively	The testing of SW on HW could also be done iterative way but there are not enough testing facilities in the company therefore, SW is designed without being tested on HW which is a challenge technically and also affects the time
Scrum		Iterative way of working is not new in HW and dividing bigger tasks into smaller tasks could be learned from the SW	not know much agile and cannot be applied to HW because in HW you cannot push any step to later phase, it has to follow a sequence	scrum could be applied to HW and should be adapted according to the needs

Themes		PL3	HW3	SW3
HW/SW development		Planning of issue is according to the waterfall approach and weekly alignment meetings with the team which considered as scrum but did not follow backlogs, sprints etc. It is based on the customer's way of working.	Do not follow a real sequence. The engineer responsible for the issue fills the template for the kick-off then it is decided to proceed with the solution or not. It is based on the customer WOW since they want to have control over the project.	SW development is done fully by scrum. It is the initiative taken by the company to follow scrum.
Project planning		Based on business case backlog, PL of the customer decide and prioritise the worklist which is done for every quarter then the PL3 define the time and the effort required for the issue in the worklist.	The PL3 assign representative for the issues and then they investigate.	Planned in Jira, and tasks are assigned based on the competencies. Also Retrospective and daily stand up are followed.
Project promises	**Time**	Defined by the customer. Quarterly plans are made but delays are possible	Not fixed, because investigation takes time and delays are not a problem. Defined based on the investigation of the issue.	Effort based estimate. The customer do have expectations, and if promised to deliver within that then it has to be done. High priority
	Cost	The overall cost is fixed, the cost of each issue is effort based, it is defined in weekly alignment with the competences.	Yearly budget, every issues should be explained and how much each issue resolution cost to the customer.	Fixed and PL3 makes the estimates based on the past experience and deduce on how much each issue cost to the customer. Good estimation needs to be made.
	Quality	High quality is required, cannot be delivered until it has an impact	Fixed and given highest priority. The customer ask to give specifications about tests done to give the guarantee that the issue is solved.	Fixed and has high priority. There is continues drive to improve the quality.
Scope		Can change but should follow scope change procedures	Keeps changing but should be fixed and defined based on the investigation of the issue.	Floating, it is effort based estimate as do not know before realizing
Overall project success	**Positive**	Yes	Yes overall successful	Mostly yes, scrum help with insight of what the current scenario of the project is.
	Negative	Team do not know what others are doing.	But between the teams there is a lack of communication that should be improved.	
Project goals and deliverables		EC is final deliverable, they are divided to form sub deliverables and defined with the competences.	Goal is described in the issue resolution issue and deliverables it depends on the issue.	Goal are deliverables are defined for each sprints. It starts with WBS by members. Lead engineers review it and then help in deciding it.
Tasks		The resource and time required for the task is defined at the beginning of issue resolution and	Defined by the responsible member. Resources and dependencies between the tasks are known at the	Teams make the WBS and make the estimates for it. It is discussed in the

		detailed estimation is done with the team and distributed in weekly meetings.	kick off.	retrospective meetings. Task are assigned based on the competency.
Requirements		the customer sets the priority. Effort estimates are made by the PL3 and competences	PL3 makes the decision. But it is not clear about the priority.	Requirement come from the customer but the PL3 makes the priority. In case of unclear requirements, it is solved in the investigation phase while making the WBS. Flexible for changing requirements.
Customer engagement		Weekly progress meetings to discuss the progress and issues in issue resolution also meetings to discuss the update of the new issue resolution	Weekly once the progress in issue is discussed with the customer but is not the part of it. The progress is communicated through the PL3	Not directly involved with the customer until there is a problem. The progress in each issue is shown to the them.
Team engagement		weekly meeting with every competence separately	Informal within the team and discussion with other teams is done only when needed. Do not generally know about what the others are working on.	Team is involved from the beginning as soon as the requirement come from the customer. The team interact and know about the work with daily stand ups.
Project evaluation		Team review once a year	No, informal discussions sometime	Yes after every sprint and at the end of the project
Project management approach	**Advantage**	The overall planning is flexible and good for the customer as they can see what is done when.	Investigation about the issues and the impact before working on it actually helps.	It represents the status of the work done by everyone and is transparent. Everyone can add value to it. Also the progress is visualised by everyone.
	Disadvantage	Using excel for planning and filling the work can be improved.	Internal problems from the customer side make the process longer. Also there is lack of communication between teams.	Some member don't like it because they think it is micro management.
Interface		There are alignment meeting, PL3 manages in case it is small otherwise the teams do it.	In the kick off it is clear. No challenges of having different way of working.	Member from the other team are invited sometime for daily stand up to follow the progress.
Scrum		Does not know much about scrum. Stand up meeting could be adapted but not daily.	Scrum does not fit for HW	Don not know about the HW fully but scrum might help. Some element of scrum should be applied.

Appendix F

	Scrum practices	***Complexity elements and problems***	***How does it solve***	***How to apply for HW***
1A	Sprint, scrum board, burndown charts	Uncertainties in scope	• Helps to see the progress, total time elapsed and track scope fulfilment. • Scope changes during sprint should be communicated with team and plan the future sprints accordingly.	• Since creating a working deliverable is difficult in HW, it is advisable to deliver something that can be discussed during the sprints. • Time constraint (Time boxed) of the sprint is not feasible therefore the length of the sprint can be flexible depending on the expected deliverable and the efforts. • Backlog of requirements for HW
		High number of project goals	• It specifies target and helps the team to work towards it. • To visualise and keep a track of all the goals and development	
		Dependency on external stakeholders	• Planning by considering the lead-time and various scenarios • Helps with an efficient and realistic plan with proper estimates	
1B	Product backlog, sprints	High number of tasks	Advices to follow one big task at a time of sprint that achievable	
		Level of competition	Helps to achieve high quality in short time with frequent release of deliverable and constant customer feedback	
2	Scrum board, daily stand-ups	Dependencies between tasks	• Help to know and visualise the dependencies, track the project status, and specify the interfaces. • Help team members to communicate about the dependencies with each other, which will improve the correlation between tasks.	Can be applied directly irrespective of the difference between HW and SW
		Involvement of different technical disciplines		
		Interfaces between different disciplines		
3	Product backlog	Insufficient/unclear requirements	• Helps to analyse where more information is needed • Help to refine and clarify the requirements.	Breaking down of the product requirements in the backlog as small as possible for HW.

	Scrum practices	***Complexity elements and problems***	***How does it solve***	***How to apply for HW***
4	Sprints, product backlog, sprint retrospectives	High project schedule drive	• help in making a realistic plan that would be achievable • help in avoiding over-commitment and false promises • aid in examining the work done throughout the project	1. applies and also • It is important to determine early in the process, which requirements need to be fixed, and which can be allowed for changes. • Use of simplicity (agile principle) should be adapted for HW, which means it lets to concentrate on actions that give importance to customer requirements.
		Changing requirements	• Help to analyse changes and control the changes, as during the sprints, changes are not accepted. • customer Involvement will help to determine and evaluate the current work	
5	Sprint planning meetings, sprint reviews/ retrospective meetings	Strict quality requirements	• The whole team together come up with solutions to solve challenges. • The team should be able to pick their tasks, which will give them a responsibility to complete their task with perfection • Help the team to analyse the work done	Can be applied directly irrespective of the difference between HW and SW
6	Daily stand-ups virtual/physical	Number of locations	• Helps to maintain close collaboration and resolve language barriers and issues with all members. • effective to convey information	• Have prior meetings • Can be applied directly irrespective of the difference between HW and SW
		Involvement of different time zones		
		Number of different nationalities		
7	Defined process/way of working according to the project	Incompatibility between different project management methods	Helps in overall technical excellence and enhance the agility.	Can be applied directly irrespective of the difference between HW and SW

Simplified version

Elements	Sprints	Scrum board	Scrum tool (burndown charts)	Product backlog	daily stand-ups	sprint retrospectives	Defined process/way of working according to the project
Uncertainties in scope	✓	✓	✓				
High number of project goals	✓	✓	✓				
Dependency on external stakeholders	✓	✓	✓				
High number of tasks	✓			✓			
Level of competition	✓			✓			
Dependencies between tasks		✓			✓		
Involvement of different technical disciplines		✓			✓		
Interfaces between different disciplines		✓			✓		
Insufficient/unclear requirements				✓			
High project schedule drive	✓			✓		✓	
Changing requirements	✓					✓	
Strict quality requirements					✓	✓	
Number of locations					✓		
Involvement of different time zones					✓		
Number of different nationalities					✓		
Incompatibility between different project management methods							✓

Appendix G

Minutes of the meeting of E1

	Scrum practices	***Complexity elements and Problems***	***Will it work?***
1A	Sprint, scrum board, burndown charts	Uncertainties in scope	Yes, it will work but I wonder if they would deliver the full solution. But for sure they will help by being clear on the information, being open using visualization, that I think these practices can bring in. It can be used in HW, It brings clarity to some ways of working that we currently use within PGP. I consider it to be supporting to the PGP activities. But No for Dependency on external stakeholders, because these practice will help to know what is going on in the project but communication and agreement with the stakeholders is what can solve this complexity. It is not the full solution for this complexity
		High number of project goals	
		Dependency on external stakeholders	
2.	Product backlog, sprints	High number of tasks	Yes, I think doing one task at time would will bring more focus and that definitely will help, because that is the issue with PGP where everything is in one phase, which can be quite confusing for the people that might be working on different task at the same time.
		Level of competition	Not specifically, because that's always in the HW project that the customer high quality in shortest time. I do not think sprints will help in achieving this complexity because it is more about scope definition and discussion. Product backlog helps, in sense it visualises the issue where you could not fully meet with the time line, it will help to make decisions. But in the end it is decision making and communication with the customer that can reduce this complexity.
2	Scrum board, daily stand-ups	Dependencies between tasks	Yes, creating visibility and in daily meetings taking decisions in providing directions to the team in different disciplines. These practices will help and reduce the complexity. but is it feasible, that is something to think, because in big projects with lot of members and disciplines, I am not sure that daily stand up with lot of members is possible/or can help and clarify on daily basis. But separate meeting within the disciplines and weekly interacting with other disciplines can work.
		Involvement of different technical disciplines	
		Interfaces between different disciplines	

	Scrum practices	***Complexity elements and Problems***	***Will it work?***
3	Product backlog	Insufficient/unclear requirements	Yes, translating requirements will help, and visualise It. It can solve and reduce this complexity but of course you need to have the discussion with customer and show them where and why some requirements needs to be specified. So the customer involvement is very much needed.
4	Sprints, product backlog, sprint retrospectives	High project schedule drive	Yes it will work if the scope is fixed.
		Changing requirements	No I am not convinced because changing can come at any time and even if you say you don't accept changes during sprints but you still have to go back after the sprints and work again so this practice will not help. But fixing some requirements at the start and doing distinction can be done for HW but at certain moment they need to be fixed and then you can't change while that's not the case for SW.
5	Sprint planning meetings, sprint reviews/ retrospective meetings	Strict quality requirements	Yes, I think so. And if teams picks the task, they will have better insight why they start certain activities or sprint sooner due to uncertainties. So involving ppl and making them pick tasks will be helpful. Currently the project leader picking for them because I think there is unclarity for some team members in how a total project will run whats their contribution.
6	Frequent communication with team and customer	Unclear tasks	Yes. It can be applied to any project even when using PGP; It is basic in project management.
7	Daily stand-ups virtual/physical	Number of locations	Yes, doing virtual meeting or talking can help in clarifying the interpretations. Even informal will work. But it can only work if there is full involvement of all members. And for the complexity with different nationality it will not work, because some nationalities are not so out spoken, so that can daily meeting not effective. So may be giving full cultural training to understand the cultural difference and using that in the all the communication will work. Having Prior meetings will work.
		Involvement of different time zones	
		Number of different nationalities	
8	Defined process/way of working according to the project	Incompatibility between different project management methods	No, because current Project management tool used in the company is already very flexible and can be adapted to all the projects. We allow different ways of working within PGP, but the complexity of incompatibility is related to problems of understanding or perception or

	Scrum practices	*Complexity elements and Problems*	*Will it work?*
			related to reaching of maturity level in following the phase.

Minutes of the meeting of E2

	Scrum practices	*Complexity elements and Problems*	*Will it work?*
1A	Sprint, scrum board, burndown charts	Uncertainties in scope	Yes, I do not see any problem in applying but All of them can also be solved from the tools of the traditional approach. However, yeah it surely will help.
		High number of project goals	
		Dependency on external stakeholders	
1B	Product backlog, sprints	High number of tasks	Yes, it can help. Product backlog helps to select tasks based on priority and give importance. One task at time is more focused and it will work. And product backlog will make the team allocation very flexible.
		Level of competition	This is about minimum lead-time, and beat the others. Which happen with clarity, well-defined steps but requires customer involvement fully
2	Scrum board, daily stand-ups	Dependencies between tasks	It can be applied; it gives a clear picture what needs to be solved. And stand-up will continuously assure and make sure you align.
		Involvement of different technical disciplines	
		Interfaces between different disciplines	
3	Product backlog	Insufficient/unclear requirements	No. I do not agree because if they are unclear, using backlog cannot help. It can only help if you have clear defined tasks. You need to work on it separately.
4	Sprints, product backlog, sprint	High project schedule drive	Yes, this can be helpful and can be applied

	Scrum practices	***Complexity elements and Problems***	***Will it work?***
	retrospectives	Changing requirements	Also here it helps yes, using sprint can check on where we go when incorporating the changes.
5	Sprint planning meetings, sprint reviews/ retrospective meetings	Strict quality requirements	No, for me a team that can do all the tasks, I do not see that in practice. The more disciplinary team, it becomes more and more difficult. You will always find some task that went wrong or someone did not do it then you will need someone else to do it outside the team. You cannot always find all competence within the team and that is my experience. The more it is one discipline then it is true. And even if you use retrospective, lack of skill can even make that practice fail.
6	Frequent communication with team and customer	Unclear tasks	Yes. I agree this helps. Although this not scrum specific solution.
7	Daily stand-ups virtual/physical	Number of locations	Yes, I agree this will help and work.
		Involvement of different time zones	
		Number of different nationalities	
8	Defined process/way of working according to the project	Incompatibility between different project management methods	I agree with this. Using one defined working is possible and based on project changes is possible.

Minutes of the meeting of E3

Scrum practices	***Complexity elements and Problems***	***Will it work?***
Sprint, scrum board, burndown charts	Uncertainties in scope	It is true what you are saying, but you should always listen to the customer. I don't understand what makes the scrum the unique solution. We have sprint way of working we just don't call it sprint, we have phases, concept and design. Sprint would not solve uncertainty in scope but help in reducing with our uncertainty to deliver. But yea it can help.

Scrum practices	*Complexity elements and Problems*	*Will it work?*
	High number of project goals	Here it can help, but it exists in most of our project. It also means we need to be able to do lot of trade-offs to achieve it.
	Dependency on external stakeholders	Yes it can help here, and that's what we try doing anyway. So it is not a unique solution.
Product backlog, sprints	High number of tasks	Limited amount of tasks to the members is helpful but yeah product backlog can help determine the tasks and determining with customer about what is the essential is more difficult
	Level of competition	very high interaction with the customer is what needed to achieve it.
Scrum board, daily stand-ups	Dependencies between tasks	Yes it is really great but the team size plays a role.
	Involvement of different technical disciplines	
	Interfaces between different disciplines	
Product backlog	Insufficient/unclear requirements	Yes to work on requirement you need to understand the requirements and also trade off plays a role here. so Discussing key requirements with the customer will work. But the selection of the key requirements are the most crucial step.
Sprints, product backlog, sprint retrospectives	High project schedule drive	I dont think just by using sprints will help working with high schedule drive there is still possibility of over committing because of the customer.
	Changing requirements	assessments can help to know what is changing but not stop changing requirements.
Sprint planning meetings, sprint reviews/ retrospective meetings	Strict quality requirements	Yes it will work, quality need to be really understood and supported by all of the members. But picking of task does not make a difference, because anyway we are all responsible.
Frequent communication with team and customer	Unclear tasks	Something we should always do. But challenge is what is good.

Scrum practices	***Complexity elements and Problems***	***Will it work?***
Daily stand-ups virtual/physical	Number of locations	I think daily stands are very good and adds value.
	Involvement of different time zones	
	Number of different nationalities	
Defined process/way of working according to the project	Incompatibility between different project management methods	Incompatibility is everywhere but does not mean having a defined way of working can manage that. But communication and understanding others work can help