Noora Pahkala

Balancing Flexibility and Structure in a NPD case company: Opportunities and Challenges of Hybrid Project Management

School of Technology and Innovations
Master's thesis
Master's Programme in Industrial Engineering and Management

Vaasa 2025

UNIVERSITY OF VAASA
School of Technology and Innovations
Author: Noora Pahkala
Title of the thesis: Balancing Flexibility and Structure in a NPD case company: Opportunities and Challenges of Hybrid Project Management:
Degree: Master of Science in Technology
Discipline: Industrial engineering and management
Supervisors: Ahm Shamsuzzoha, Petri Helo & Ville Tuomi
Year: 2025 **Pages:** 118

ABSTRACT:
Hybrid project management methodologies are increasingly relevant to organizations balancing with the flexibility of Agile practices and the structure of traditional project management practices. However, their effectiveness is shaped by the organizational context in which they are implemented and while theories exist, empirical studies remain limited. The aim of this study is to examine what challenges the current hybrid project management model has and how project success can be improved by addressing these challenges in a case company executing large and complex new product development projects. Furthermore, this study aims to recognise how early warning indicators can create transparency to help leadership identify critical project delays to enable timely actions. To examine these re-search questions, mixed methods research was conducted in a global technology company's new product development project organization. The empirical data consists of 17 semi-structured interviews working in project management roles in research and development- and supply chain departments. While the current hybrid model was considered as the most effective project management method compared to previous ones, the findings reveal several structured challenges in the current implementation of the hybrid model, including short-term focus, lack of overall management, unclear roles and insufficient transparency between stakeholders. These factors are closely linked to frequent project delays resulting from resource constraints and unrecognized project tasks and their dependencies. Strengthening change management, common planning, aligning schedules and resources to realistic levels, and increasing flexible resource use are identified as key actions for improving project performance. Furthermore, improved recognition of tasks and their dependencies for critical path monitoring and transparency through common tools are identified as key actions for improving detection of early warning indicators and supporting overall project success.

KEYWORDS: Project management, Agile, Hybrid, Leadership, Management

VAASAN YLIOPISTO
Tekniikan ja innovaationjohtamisen akateeminen yksikkö
Tekijä: Noora Pahkala
Tutkielman nimi: Balancing Flexibility and Structure in a NPD case company: Opportunities and Challenges of Hybrid Project Management:
Tutkinto: Diplomi-insinööri
Oppiaine: Tuotantotalous
Työn ohjaajat: Ahm Shamsuzzoha, Petri Helo & Ville Tuomi
Valmistumisvuosi: 2025 **Pages:** 118

TIIVISTELMÄ:
Hybridiprojektihallintamenetelmät ovat yhä merkityksellisempiä organisaatioille, jotka pyrkivät tasapainottelemaan ketterien menetelmien joustavuuden sekä perinteisten projektinhallintamenetelmien rakenteen ja hallinnan välillä. Projektihallintamenetelmien tehokkuus riippuu kuitenkin organisaation kontekstista, jossa niitä sovelletaan. Vaikka hybridiprojektinhallintamenetelmistä on olemassa teoreettista tutkimusta, empiirisiä tutkimuksia on edelleen vähän. Tässä tutkimuksessa tarkastellaan, mitä haasteita nykyisessä hybridiprojektinhallintamallissa on ja miten näihin haasteisiin keskittymällä voidaan parantaa projektimenestystä case-yrityksessä, joka toteuttaa isoja ja monimutkaisia uustuoteprojekteja. Lisäksi tutkimuksessa pyritään tunnistamaan, miten varhaisvaroitusindikaattorit voivat lisätä läpinäkyvyyttä ja auttaa johtoa havaitsemaan kriittiset projektiviiveet ajoissa. Tutkimuskysymysten tarkasteluun käytettiin sekamenetelmää ja empiirinen tutkimus toteutettiin globaalin teknologiayrityksen uustuoteprojektien osastolla. Empiirinen aineisto koostuu 17 puolistrukturoidusta haastattelusta, joissa haastateltavat työskentelevät projektihallinnan tehtävissä tutkimus ja kehitys - sekä toimitusketjuosastoilla. Vaikka nykyistä hybridimallia pidettiin tehokkaimpana projektinhallintamenetelmänä aiempiin verrattuna, tulokset paljastavat useita rakenteellisia haasteita nykyisessä hybridimallissa, kuten lyhytnäköisen keskittymisen, puutteellisen kokonaisjohtamisen, epäselvät roolit ja riittämättömän näkyvyyden sidosryhmien välillä. Nämä haasteet liittyvät läheisesti toistuviin projektiviiveisiin, joita aiheuttavat erityisesti resurssirajoitukset sekä tunnistamattomat tehtävät ja niiden riippuvuudet. Muutoshallinnan ja yhteisen suunnittelun vahvistaminen, aikataulujen ja resurssien realistinen kohdentaminen sekä resurssien joustavamman käytön lisääminen tunnistetaan keskeisiksi toimenpiteiksi projektien onnistumisen parantamiseksi. Lisäksi tehtävien ja niiden riippuvuuksien parempi tunnistaminen kriittisen polun seurantaa varten sekä näkyvyyden lisääminen yhteisten työkalujen avulla tunnistetaan keskeisiksi toimenpiteiksi varhaisvaroitusindikaattorien havaitsemisen parantamiseksi sekä projektien kokonaisvaltaisen onnistumisen tukemiseksi.

AVAINSANAT: Projektinhallinta, Ketterät menetelmät, Hybridi, Johtaminen

Contents

1 Introduction 8

1.1 Case company 8

1.2 Goals and thesis structure 10

2 Literature review 12

2.1 Agile Project Management 12

2.1.1 Agile in NPD and hardware projects 25

2.1.2 Strengths and Weaknesses of Agile and Scrum 32

2.2 Traditional project management & Waterfall model 41

2.2.1 Stage-Gate model 42

2.2.2 Strengths and weaknesses 45

2.3 Hybrid project management 49

2.3.1 Strengths and challenges of hybrid project management 53

2.4 Project success 60

3 Methodology 62

3.1 Mixed methods research 62

3.1.1 Qualitative research 63

3.1.2 Quantitative research 63

3.1.3 Data collection 64

3.2 Data analysis 66

3.2.1 Thematic analysis 66

3.2.2 Quantitative analysis 68

4 Results 69

4.1 Project management methodology 69

4.2 Project initiation and clarity of goals 70

4.3 Scope definition and changes 74

4.4 Task planning and critical path 75

4.5 Project schedule 79

4.6 Project delays 83

4.7 Multiteam collaboration 86

4.8 Hybrid project management methodology challenges 87
4.9 Hybrid project management methodology strengths 91
5 Discussion 96
5.1 Key findings 96
5.1.1 Main challenges of current hybrid project management model 97
5.1.2 Early warning indicators and reaction to project delays 101
5.2 Managerial recommendations 102
5.3 Limitations and future research 105
6 Conclusions 107
References 109
Appendix 1. Interview questionnaire 117

Figures

Figure 1. APM iterations (adapted from Breyter, 2023) 17
Figure 2. Sprint events 23
Figure 3. Waterfall model (adapted from Breyter, 2023) 42
Figure 4. Stage-Gate model (adapted from Cooper, 2023) 43
Figure 5. Agile-Stage-Gate model (Cooper & Sommer, 2018) 53
Figure 6. Strategic planning process (adapted from case company process chart) 71
Figure 7. Milestone deliverables plan (adapted from case company's milestone plan) 72
Figure 8. Clarity of the project objective in the beginning 73
Figure 9. Commonness of scope changing during the project 75
Figure 10. M3 from SCRS list (example project from case company) 76
Figure 11. Commonness of project tasks changing during the project 78
Figure 12. Commonness of projects staying on schedule 80
Figure 13. Successfulness of the current PMM 95

Tables

Table 1. Strengths of APM 35
Table 2. Challenges of APM 40
Table 3. Strengths of TPM 46
Table 4. Challenges of TPM 48
Table 5. Strengths of HPM 55
Table 6. Challenges of HPM 59
Table 7. Interview data 64
Table 8. Reasons for project delays 85
Table 9. Challenges of the current PMM 90
Table 10. Strengths of the current PMM 93

Abbreviations

APM	Agile project management
ART	Agile release train
CMMI	Capability maturity model integration
HIP	Hardening, innovation, planning
HPM	Hybrid project management
IPAC	Iteration, prototype, alignment & customer
ISO	International organization for standardization
LeSS	Large-scale Scrum
MAHD	Modified Agile for hardware
NPD	New product development
PM	Project management
PMM	Project management methodology
PO	Product owner
R&D	Research & development
SAFe	Scaled Agile Framework
SC	Supply chain
SIPOC	Supplier, input, process, output, customer
TPM	Traditional project management
TPO	Technical product owner

1 Introduction

The nature of project management is undergoing a fundamental transformation. Traditional project management (TPM) methodologies were originally developed to support large and linear projects with stable environments and predictable outcomes. However, because today's projects are often executed in increasingly complex and dynamic contexts, characterised by rapid technological developments and increased uncertainty, TPM has proven to be too rigid and slow (Cooper, 2016; Zasa et al., 2021). In response to these changes, organizations have increasingly turned to Agile methodologies, which was developed in software industry, emphasizing adaptability, responsiveness and fast iterations of development (Ciric et al., 2018; Trivedi, 2021, Zasa et al., 2021). However, organizations in other industries aiming to adopt Agile project management (APM) often struggle with the methodology's lack of structure and governance (Sithambaram et al., 2021; Zasa et al., 2021). This is particularly the case in large organizations that execute large scale and complex projects, working in fields with strict regulations and governance (Sithambaram et al., 2021; West et al., 2021; Zasa et al., 2021). As a result, many organizations have begun to integrate Agile and TPM methods through hybrid project management (HPM) methodologies, combining the flexibility of Agile practices with the structure provided by traditional practices (Cooper, 2016; Cooper & Sommer, 2018; Reiff & Schlegel, 2022; Sommer et al., 2015; Székely et al., 2025; Zasa et al., 2021). Despite their growing adoption, the successful implementation of HPM models remains challenging (Bianchi et al., 2020; Cooper & Sommer, 2018; Zasa et al., 2021), since APM and TPM offer in many ways contradictory guidelines. Moreover, empirical research on how these methods are implemented in practice is still limited, and their effectiveness is strongly shaped by the organizational context in which they are applied.

1.1 Case company

The case company is a global technology organization specializing in advanced solutions designed to reduce energy consumption, increase machine productivity and enable electrification to lower overall emissions. This study focuses on a business segment

dedicated to power conversion and energy efficient solutions. More specifically, this study is conducted for a new product development (NPD) project organization, which is responsible for driving innovation and delivering new products to the market. The NPD project organization has adopted a HPM methodology, combining TPM practices through the Stage-Gate model with Agile practices through utilizing Scrum framework in daily work.

This study includes two key departments involved in executing NPD projects: research and development (R&D) and supply chain (SC). R&D department consists of product teams, which own the products and are responsible for product design and its development. SC department is responsible for developing the whole supply chain for new products, from physical assembly lines to system related maturity. SC department is a shared resource, working for multiple product teams simultaneously. While these two departments work closely together to execute NPD project, they have different organizational structures. R&D department operates in Agile organization with product teams that involve the roles of product owner (PO), technical product owner (TPO) and scrum master. Whereas the SC department follows a more traditional organizational structure with the role of a project manager and team members assigned to specific projects. TPO is not officially part of a product team in Agile, but the case company has decided to include it to support PO and developers. While PO works more on customer interface, TPO works mostly towards the product team and developers to break down product goals into practical things that the product team can deliver. In practice, TPO is an experienced and technically competent team member that works between PO and the customer, and the developers. This organizational setup provides a relevant context for this study.

NPD projects are organized so that there are global product platform projects that include a whole project family and multiple different product variants. These platform projects are extremely large and are multiple years long, involving well over hundred people. From the product platform, the productization of individual products and variants are considered as their own projects, lasting around one to two years, involving around

thirty people. Through the product platforms, the productization projects of products and variants can be executed faster since same materials are utilized as much as possible.

These NPD projects are managed with the Stage-Gate model but daily work is organised with Scrum framework, meaning that a HPM model is used. The case company's NPD project organization has transitioned from a TPM methodology to a HPM relevantly recently during the past few years and further development is required to ensure successful project execution. In addition, the case company frequently faces project delays and has a need to improve the transparency of early warning indicators for leadership to enable timely and informed actions.

1.2 Goals and thesis structure

The goal of this study is to address the key challenges related to the successful adoption of a HPM methodology and project delays in the case company. While theoretical studies of HPM methodologies have been conducted, empirical research remains limited. Furthermore, the context in which project management methodologies (PMM) are implemented significantly influences its effectiveness, making it essential to study the current adoption of a HPM methodology within the context of the case company and the specific challenges it faces. In addition to the effective adoption of HPM methodology as a critical success factor, identifying early warning indicators to create transparency of the actual project status is also essential.

To address these specific challenges in the case company the following research questions are posed:
RQ1: What are the challenges of the current hybrid project management model that should be addressed to improve project success?
RQ2: What early warning indicators create transparency to help leadership identify critical delays in projects?

To address these two research questions, first a theoretical basis for the study is presented in literature review. The literature review presents relevant project management methodologies as well as the benefits and challenges related to them. First, overviews of Agile and TPM are presented separately and then the combination of these is explored through HPM. Furthermore, a chapter discussing project success in NPD projects is included for taking the characteristics of complex projects and their effect on the perspectives of project success into account. Moving to the empirical study part of the thesis, first research methodology and execution of the study are presented. This is followed by the results, which are then interpreted and discussed in relation to previous literature. Finally, managerial recommendations are presented, along with the limitations of the study and suggestions for future research.

2 Literature review

Project management institute (2021) defines project as a temporary venture undertaken to create a unique product or service in a predetermined timeframe which has a clear beginning and end. Projects can either be stand-alone or part of a program or portfolio and they are expected to create value for the organization (Project Management Institute, 2021). Projects can create value in different ways, for example through business benefits, new product development, system development or investment in company infrastructure (Wells, 2012). Kodukula (2014) highlights that only projects that align with the strategy and goals of the organizations are selected and executed. Project management is defined as the application of knowledge, skills, tools, and techniques to project activities to meet project requirements (Project Management Institute, 2021). The purpose of project management is to guide the project work to deliver the intended outcomes and results within the project requirements (Project Management Institute, 2021). Many different project management methodologies have been developed to help with successful project management and project execution. Project management methodology is defined as a structured and systematic set of principles, practices, tools and techniques that guide the planning, execution, monitoring and control of a project (Spundak, 2014; Unqureanu & Uncureanu, 2014).

In the following chapters project management approaches, methodologies and models that are relevant in the context of the case company are presented. The strengths and weaknesses of each are reviewed individually. Then a HPM methodology is presented to review how two different project management methodologies can be combined to gain the strengths of both. In addition, the definition of a project success is presented in the context of complex large-scale projects.

2.1 Agile Project Management

Agile approaches have grown to be part of the most common mainstream project management approaches during the past two decades (Gemino et al., 2021). Agile Manifesto

was published in 2001 specifically for software development and it has been generally accepted as a foundation for APM (Ciric et al., 2018; Ciric Lalic et al., 2022). Cobb (2023) writes that Agile Manifesto combined clearly all previous Agile methodologies into specific values and principles. Agile Manifesto consists of four values and twelve principles to backup those values (Beck et al., 2001). Ciric Lalic et al. (2022) write that Agile approach aims to promote deeper understanding of software development project complexity. The four values of Agile manifesto are:

1. Individuals and interactions over processes and tools.
2. Working software over comprehensive documentation.
3. Customer collaboration over contact negotiation.
4. Responding to change over following a plan (Beck et al., 2001).

Beck et al. (2001) explain that the four values of Agile Manifesto have been formulated in a way that while there is value on the right side of the statement the left side of the statement is valued more. The first statement means that Agile ultimately values intelligent decision-making by skilled and empowered people and interactive collaboration as a team (Cobb, 2023). This comes from an essential element of Agile approach that is the distribution of responsibility between the project team members and the team being self-organizing (Gemino et al., 2021). Furthermore, this means that in Agile approach project management has a more softer leadership approach with flexible and adaptive processes instead of control-oriented and highly directive project management approach with strict processes (Cobb, 2023). Cobb (2023) highlights that this doesn't mean that Agile approach wouldn't have well-defined processes but that they are designed to be adaptive and meant to be implemented intelligently instead of following certain prescriptive processes mechanically. In addition, tools are kept in a supportive role to facilitate human interaction (Cobb, 2023).

The second statement means that documentation itself should not be an essential value but rather it should always provide value to the project (Cobb, 2023). According to Cobb

(2023), this was a response to the Waterfall model, in which documentation is over emphasized. Agile's more flexible approach to documentation addresses the challenge many users face in defining detailed requirements early in a project (Cobb, 2023). This is a significant challenge especially in uncertain and changing environment (Cobb, 2023). Cobb (2023) further explains that by reducing dependency on extensive documentation, the risk of miscommunication and misunderstandings about the intended requirements are also minimized.

The third statement describes the value of a collaborative approach over strictly defined contracts. This is also a response to the Waterfall model, which often centers around the triple constraint of scope, schedule and cost (Cobb, 2023; Kodukula, 2014). According to Kodukula (2014), this focus on controlling the requirements of the contract may overlook broader indicators of project success, such as customer satisfaction and long-term stakeholder value. Agile addresses this limited view by emphasizing customer collaboration, encouraging a more adaptive and trust-based relationship with customers (Cobb, 2023). Cobb (2023) explains that this enables creating a more general agreement with high-level requirements and leave flexibility to define detailed requirements during later stages of a project. In uncertain environment this approach supports value creation beyond rigid contract specifications, recognizing the challenge of defining detailed requirements in the beginning of a project (Cobb, 2023). Cobb (2023) highlights the importance of always adapting the Agile approach based on the situation and finding a right balance with a customer to agree on a suitable contract.

The fourth, and final statement, addresses the difficulty of defining requirements at the start of a project and recognizes that it's often not effective or realistic (Cobb, 2023). According to Thesing et al. (2021), users aren't often able to specify the details of general requirements, or the steps to achieve the defined objectives might not be clear. Cobb (2023) writes that in these situations it's suitable to leave some space in the project plan for the detailed requirements to change during the project. Agile approach provides

flexibility to respond to changing situations instead of strictly focusing on following a predefined project plan (Cobb, 2023).

Lowell (2023) writes that the values highlight that it's more important to make progress than it's to make something perfect. The center of the four values is individuals and collaboration between them in order to respond to change flexibly (Lowell, 2023). According to Cobb (2023), all four values are interrelated and need to be contemplated to find a suitable approach to a particular project. Furthermore, Cobb (2023) emphasizes that there are a lot of alternatives to all-or-nothing decisions and finding the right balance of control and flexibility is important.

In addition to the four values that form the foundation of Agile, Beck et al. (2001) also defined twelve principles to expand the values and provide more detail. Cobb (2023) highlights that as with the four values, these principles should not be considered as absolutes but more context-depended preferences. The twelve principles of agile in an adapted formulation are:

1. The highest priority is to satisfy the customer with quick and continuous delivery of valuable software.
2. Embrace evolving requirements, even late in development to provide competitive advantage to customer.
3. Deliver working software frequently in short iterations of couple of weeks to couple of months.
4. Daily communication and collaboration between businesspeople and developers is required.
5. Create a space for motivated people, offer guidance and resources and trust their ability to succeed.
6. Face-to-face conservation is the most efficient and effective form of sharing information to and within a development team.
7. The primary measure of progress is working software.

8. Sustainable and manageable work rates are supported so that sponsors, developers and users can keep a consistent pace indefinitely.
9. Agility is enhanced with continuous attention to technical excellence and quality design.
10. The amount of work not done is maximized for simplicity.
11. Self-organizing teams are optimal for achieving best architectures, requirements and designs.
12. The team regularly reflects and adjusts their own ways of working to become more effective and efficient (Beck et al., 2001).

These values and principles of Agile Manifesto provide a strong foundation for understanding Agile (Cobb, 2023). Furthermore, Cobb (2023) highlights that many of the values and principles can be applied to almost any project when done intelligently and in the right context.

Fundamentally, **APM** is about succeeding in changing environments by identifying and understanding the uncertainty and being able to adapt and respond to change quickly (Trivedi, 2021). As Ciric et al. (2018) claim, the companies that will succeed in the quickly changing world of manufacturing are those that have developed the agility required to support rapid and continuous innovation. Gemino et al. (2021) writes that the ability to adapt and respond to change comes from dividing the project work into separate iterative cycles in which the project and it's requirements are continuously further refined. According to Breyter (2023) each iteration contains the process of the Waterfall model: initiation, planning, execution, monitoring and controlling and closing. The difference to the Waterfall model is that value is delivered with each short iteration, allowing customer value to be realised much earlier in the project rather than only at the end of the project (Breyter, 2023). The lifecycle of APM is shown in Figure 1, where customer releases are done after each iteration after the minimum viable product is ready (Breyter, 2023).

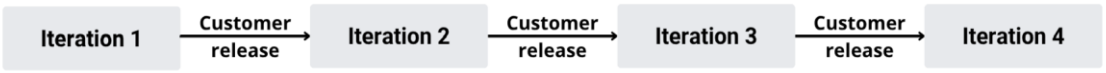

Figure 1. APM iterations (adapted from Breyter, 2023)

In addition to early value delivery, short iterations are what makes APM flexible and adaptive. Breyter (2023) writes that in APM the product is delivered gradually with each iteration which enables the users to give feedback on the product throughout the project. When feedback is provided after each iteration the project team can adapt and respond to the feedback quickly during the upcoming iterations (Breyter, 2023). These iterative cycles require a shift in the approach of project planning from controlling changes in scope, cost and schedule to adopting a more adaptive approach focused on providing value (Cobb, 2023).

Another essential element of APM is that the project team is self-organizing and responsibility is divided between the team members (Cobb, 2023; Gemino et al., 2021; Hoda & Murugesan, 2016). According to Hoda et al. (2012) self-organizing teams improve the organization's flexibility and ability to respond to change which makes it a crucial element of APM. Hoda et al. (2012) emphasize that self-organizing teams are one of the critical success factors of Agile projects. Furthermore, Cobb (2023) writes that this means that Agile processes largely depend on skilled and empowered team members who can plan and organise their own work independently based on agreed goals without detailed directions given to them. Ultimately, self-organizing teams must have mutual trust and respect, a common focus and the ability to repeatedly re-organize to meet new challenges (Hoda et al., 2012). In addition, Lawong and Akanfe (2025) write that team dynamics are essential for achieving effective project management. Work role performance is built from three dimension of work role performance: proficiency, proactivity and adaptivity. As stated previously, achieving these three dimensions is crucial for self-organizing teams to achieve effective APM.

There are six different characteristics of a self-organizing team: autonomy, shared leadership, team orientation, communication, redundancy and learning (Hoda & Murugesan, 2016; Perlak, 2019). Hoda & Murugesan (2016) write that self-organizing teams have high level of autonomy that is enabled by the support from management. Management provides minimal critical specifications with high-level goals and direction but leaves everyday decision-making to the team (Hoda & Murugesan, 2016). According to Hoda & Murugesan (2016) autonomy in self-organizing teams can be viewed across three different dimensions: individual, internal and external. Individual autonomy refers to each team member's ability to manage their own tasks and responsibilities (Hoda & Murugesan, 2016). Internal autonomy of the team concerns the extent to which decisions are made collectively within the team instead of by centralized decision-making by an individual leader (Hoda & Murugesan, 2016). External autonomy involves the degree of independence the team has from external stakeholders, including management, in determining how to achieve the agreed goals (Hoda & Murugesan, 2016).

Because self-organizing teams are designed to operate with a high degree of independence, the responsibility of decision-making and execution which is traditionally held by project managers is shared with the team (Hoda & Murugesan, 2016; Hoda et al., 2012). According to Hoda & Murugesan (2016) this mean that in addition to executing technical tasks, Agile team members are expected to share responsibilities in project management activities such as planning estimation, stakeholder collaboration, task allocation and progress tracking. By bringing decision-making authority closer to the operational level and involving Agile teams in management practices, organizations aim to improve the speed and accuracy of problem-solving. However, Hoda et al. (2012) emphasize that autonomy doesn't mean leadership doesn't exist in APM. Instead, leadership in self-organizing teams is characterized by a light-touch and adaptive approach that focuses on guiding rather than controlling (Hoda et al., 2012, Perlak, 2019). In APM, leaders are still responsible for critical functions such as setting direction, motivating teams, aligning efforts and ensuring the availability of resources (Hoda et al., 2012). Perlak (2019) writes that

managers serve as facilitators who create a supportive environment, share relevant information and intervene only when necessary to help the team.

Hoda et al. (2012) explain that a truly autonomous team has both freedom and responsibility with minimal interference in day-to-day operations by management. This level of trust encourages self-monitoring, a strong sense of ownership and increased commitment among team members (Hoda et al., 2012). Furthermore, in mature Agile teams, autonomy often evolves into self-transcendence, a state in which team define their own goals and continuously evaluate and improve their methods for achieving them (Hoda et al., 2012). Hoda et al. (2012) explains that self-transcendence teams may determine the work load they can commit to in each iteration and take full responsibility for delivering that commitment, thereby reinforcing both internal motivation and performance pressure.

One other essential element of APM is close and continuous communication between stakeholders through both formal and informal communication (Gemino et al., 2021). Collaboration, communication and status reporting have been identified as critical elements of project success in APM (Boerman et al., 2015; Pikkarainen et al., 2008). These elements are important since in APM detailed requirements are defined as the project unfolds, which requires a collaborative and trust-based partnership between the customer and project team (Cobb, 2023). Furthermore, Boerman et al. (2015) write that effective status reporting to external stakeholders is required to facilitate communication, build confidence and provide appropriate project steering. Because APM requires active participation from external stakeholders, pure Agile approach won't be suitable if the customer cannot or doesn't want to actively engage in the project (Cobb, 2023).

APM emphasizes frequent face-to-face interaction to enhance mutual understanding and accelerate decision-making processes across stakeholder groups (Pikkarainen et al., 2008). In a study conducted by Pikkarainen et al. (2008), practices such as open office spaces and daily meetings were effective in facilitating communication of project goals,

status updates and decisions within project teams. These informal communication practices not only enhanced collaboration but also reduced the need for excessive documentation, allowing more flexibility to respond to change (Pikkarainen et al., 2008).

However, while informal communication methods can increase efficiency, Boerman et al. (2015) caution that relying too heavily on tacit, face-to-face communication may become problematic in large or complex environments. As complexity grows communication may reach its limits, making it more difficult to ensure that critical information reaches all relevant stakeholders in a timely and accurate manner (Boerman et al., 2015). This underlines the importance of balancing informal communication with formal status reporting, especially when coordinating across distributed teams or departments (Boerman et al., 2015).

Scrum is the most widely used Agile framework (Breyter, 2023; Cobb, 2023; Zavyalova et al., 2020). Schwaber and Sutherland (2020) have written the Scrum Guide in which they define Scrum as a lightweight framework designed to deliver value in complex environments though adaptive solutions. Schwaber and Sutherland (2020) write that Scrum is based on empiricism and lean thinking. In empiricism, knowledge comes from experience and decision are made based on observations (Schwaber and Sutherland, 2020). In lean thinking waste is reduced and focus is on the essentials (Schwaber and Sutherland, 2020). Furthermore, scrum is built on three interrelated pillars: transparency, inspection and adaption (Breyter, 2023; Schwaber & Sutherland, 2020). Transparency means that emergent process and work must be transparent to all internal and external stakeholders. Transparency on the project work increases the risk of value decreasing decisions and other risks (Lawong & Akanfe, 2025; Schwaber & Sutherland, 2020). Transparency enables inspection that must be done frequently to identify potential problems (Schwaber & Sutherland, 2020). Inspection further enables adaption when it's needed to keep the project on the right track (Schwaber & Sutherland, 2020). Schwaber and Sutherland (2020) explain that the Scrum framework is intentionally left incomplete to leave

flexibility for people to implement it in the most suitable way. This incompleteness may be partly why Breyter (2023) describes Scrum as simple to understand but difficult to master.

Scrum framework can be divided into three categories: roles, events and artifacts (Lawong & Akanfe, 2025). Scrum roles are designed for a small team of people and only consists of one scrum master, one product owner (PO) and developers (Schwaber & Sutherland, 2020). Lawong and Akanfe (2025) note that developers are often referred as simply team member outside software industry. The scrum team is responsible for all product related activities and accountable for creating value (Schwaber & Sutherland, 2020). A scrum team should be small, maximum ten people, to maintain efficient communication, productivity and adaptivity but large enough to have necessary expertise and skills to complete the project (Lawong & Akanfe, 2025; Schwaber & Sutherland, 2020). The project size is not a reason to grow the scrum team but instead the team should be reorganized into multiple teams that are focused on the same product (Gustavsson & Rönnlund, 2013; Schwaber & Sutherland, 2020)

Scrum master ensures that Scrum is understood and properly executed within the team and organization (Cobb, 2023; Schwaber & Sutherland, 2020). Scrum master acts as a servant-leader who supports improvement, removes obstacles, coaches members and promotes effective scrum practices while also serving the organization by promoting Scrum theory and leading its adopting (Breayter, 2023; Schwaber & Sutherland, 2020). Cobb (2023) highlights that although scrum master provides guidance, scrum teams are expected to be as self-organizing as possible. PO is responsible for the product vision and maximizing product value by managing the product backlog and facilitating open stakeholder communication (Lawong & Akanfe, 2025; Schwaber & Sutherland, 2020). PO represents stakeholders, makes final backlog decisions, prioritizes features and ensures that goals and backlog items are clearly communicated to the scrum team (Lawong & Akanfe, 2025; Schwaber & Sutherland, 2020). Developers, or team members, plan and adapt sprint goals daily, deliver usable product increments each sprint and ensure quality

through the definition of Done (Breyter, 2023; Cobb, 2023; Schwaber & Sutherland, 2020). Lawong and Akanfe (2025) emphasize the importance of having a diverse and complementary skill set among the team members

Scrum events are all included in iterations that are called sprints in Scrum (Breyter, 2023; Lawong & Akanfe, 2025). Sprints are continuous fixed length iterations of four week or less during which value is provided (Breyter, 2023; Cobb, 2023; Schwaber & Sutherland, 2020). Schwaber & Sutherland (2020) explain that one sprint contains all other scrum events: sprint planning, daily scrums, sprint review and sprint retrospective (Figure 2). According to Schwaber & Sutherland (2020) these sprint events provide the transparency that is required, thus minimizing the need for any other meetings during the sprint. In addition, through transparency the events also allow for inspection and adaption (Lawong & Akanfe, 2025). During each sprint the goal is to produce a result with functionality that is useful for the final product (Loweng & Akanfe, 2025). Gustavsson and Rönnlund (2013) explain that because each sprint is fixed length, the length of the sprint supersedes the tasks planned for the sprint regardless of the stage of the tasks and if they have been completed. This is called a time-box concept and furthermore it means that the product owner and developers must continuously prioritize the tasks that should be done for the sprint (Gustavsson & Rönnlund, 2013).

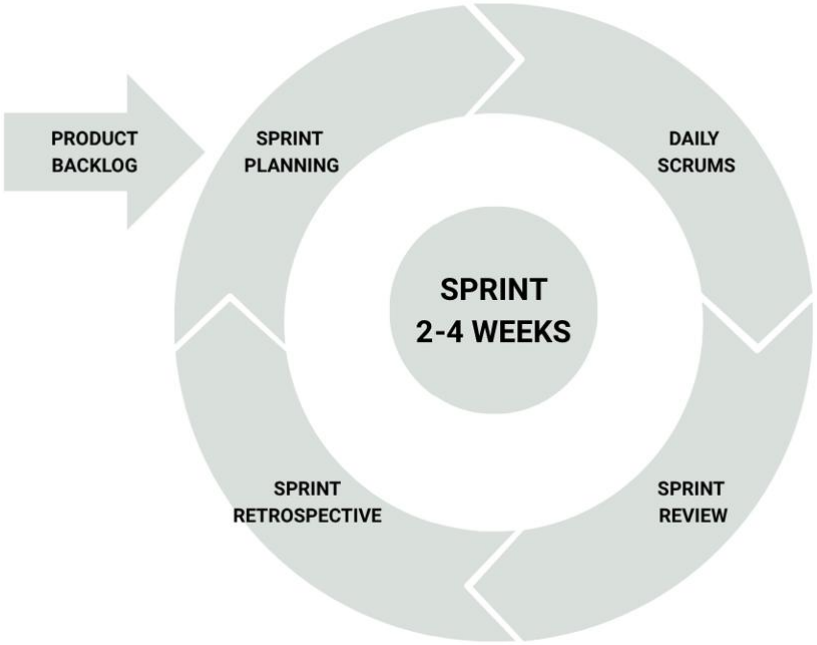

Figure 2. Sprint events

Sprint planning divides project work into small and manageable pieces that can be completed within one sprint, setting goals and defining the work collaboratively with the scrum team (Cobb, 2023; Schwaber & Sutherland, 2020). Schwaber and Sutherland (2020) explain that sprint planning is for determining the definition of Done and addressing key questions about the sprint's value, achievable work and approach to do the work. The definition of Done represent a clear and shared understanding of the criteria that must be met for a product backlog item to be considered complete and usable (Lawong & Akanfe, 2025; Schwaber & Sutherland, 2020). Throughout the sprint, short 15-minute meetings called daily scrums are held to review progress, discuss completed and planned work, identify challenges and adapt plans when necessary (Lawong & Akanfe, 2025: Schwaber & Sutherland, 2020). According to Schwaber and Sutherland (2020), daily scrums are important for improving communication, problem-solving and promoting fast decision-making among the scrum team, although team members also often have more detailed discussions outside of daily scrums. At the end of the sprint, the scrum team holds a sprint review with key stakeholders to present outcomes, gather feedback and plan future adaptions (Lawong & Akanfe, 2025; Schwaber & Sutherland, 2020). It is

emphasized by Schwaber and Sutherland (2020) that sprint review should not only be a presentation but a working session. While sprint review focuses on the product and results of the sprint, sprint retrospective focuses on the process to complete the work (Lawong & Akanfe, 2025). The purpose of sprint retrospective is to find ways to improve processes, quality and team effectiveness through reflecting on tools, interactions and challenges faced (Schwaber & Sutherland, 2020). Sprint retrospective is the event that concludes the sprint.

The third category of scrum framework is scrum artifacts that represent work or value, containing artifacts of the product backlog, the sprint backlog and the increment (Schwaber & Sutherland, 2020). These artifacts promote transparency, shared understanding and focus by providing a reference point for measuring progress (Lawong and Akanfe, 2025; Schwaber and Sutherland, 2020). The product backlog is developed and refined by PO with inputs from key stakeholders and the scrum team and it's continuously updated with detailed and prioritized items (Lawong & Akanfe, 2025; Schwaber & Sutherland, 2020). Schwaber and Sutherland (2020) explain that the product backlog is linked to the product goal, which defines the future state of the product and the product goal must be either fulfilled or abandoned before starting a new one. Developers select items from the product backlog to form the sprint backlog for each sprint, refining them with details such as priority and difficulty, and updating them throughout the sprint to track progress towards the sprint goal (Lawong & Akanfe, 2025; Schwaber & Sutherland, 2020). Schwaber and Sutherland (2020) explain that the sprint goal is flexible in a way that if the work proves to be different from what was initially expected, the developers can work with the product owner to adjust the scope of the sprint backlog while maintaining the sprint goal. The final artifact, the increment, represents a verified and usable addition to the product that contributes to the product goal and must meet the determined definition of Done.

Because scrum framework has been extensively successful, many larger enterprises with large-scale projects have also adopted scrum framework. However, since scrum

framework was designed to small teams, implementing scrum to large-scale projects is not straightforward. Large-scale projects have greater complexity and require more resources consisting of multiple teams (Almeida & Espinheira, 2022). To modify Scrum framework to better support the characteristics of large-scale projects, Larman and Vodde developed **large-scale scrum framework (LeSS)** (Almeida & Espinheira, 2022). The fundamentals of Scrum framework: scrum events, iterative development, adaptability and transparency are the same in LeSS, but additionally guidelines for organizational structure, product management and working with multiple teams are presented (Almeida & Espinheira, 2022; Alqudah & Razali, 2016; Uludag et al., 2019). LeSS supports scalability by using a single product owner and central product backlog to facilitate coordination between multiple teams (Alqydah & Razali, 2016; Uludag et al., 2019). This is supported by modified coordination practices of joint scrum events such as sprint planning, backlog refinement and retrospective which are attended by limited team representatives to enhance efficiency and alignment across teams (Alqudah & Razali, 2016; Uludag et al., 2019). Furthermore, according to Almeida and Espinheira (2022), LeSS encourages organizational simplification with a minimal set of rules and a shift from project-oriented mindset to a product-oriented mindset. Additionally, LeSS presents ten technical and development practices to ensure high quality and adaptability in large-scale delivery: acceptance tests, architecture and design, clean codes, continuous delivery, continuous integration, specification by example, test-driven development, test-driven thinking, test automation and unit tests (Almeida & Espinheira, 2022). Almeida and Espinheira (2022) emphasize that empirical process control, minimizing role complexity and maintaining customer focus through incremental delivery, LeSS aims to scale agility without compromising the core values of Scrum.

2.1.1 Agile in NPD and hardware projects

Although APM is still primarily used in software development and lacks extensive theoretical or practical grounding outside of it, according to Ciric et al. (2019) there is growing evidence that its use is increasing in other industries as well. APM is being increasingly applied also in hardware development due to it being particularly suitable for uncertain

and complex conditions in which hardware development projects are often executed nowadays (Atzberger & Paetzold, 2019; Zasa et al., 2021). Ciric et al. (2019) write that the key motivations for adopting APM across different sectors include the need to decrease the time for product or project delivery and improve responsiveness and adaptability to shifting priorities. However, Simpson and Sheth (2025) state that the pitfalls of trying to apply software-based Agile methods to physical products is that the tactics are not designed for hardware and critical elements are missing, which often leads to poor results. Moreover, while Gustavsson and Rönnlund (2013) note that understanding the behaviour of hardware products require extensive iterative work and simulated testing similar to software development, there are still several challenges unique to hardware development that must be carefully considered when applying APM in that field. Overall, according to Atzberger et al. (2019) the benefits of implementing APM are underestimated in soft factors such as communication, transparency and commitment, but on the other hand, overestimated in hard factors of controlling cost, time and quality.

Cobb (2023) has recognized that the standard training for scrum masters and product owners focuses more on the theory and mechanics on how to practice scrum than on how to apply it in different situations such as hardware development. According to Cobb (2023), using Agile in hardware development does not necessarily mean directly applying frameworks like Scrum, as often done in software development. Instead, it often involves creatively adapting Agile and Lean principles to build a highly efficient and effective development process suited to the needs of hardware development (Cobb, 2023). Weichbroth (2022) also highlights that agile practices must be adapted to fit specific contexts and that APM presents its own challenges, risks and limitations in hardware development. The reason why adopting Agile to hardware development requires careful consideration is because hardware development has unique challenges, such as constraints of physicality, which significantly differentiate it from software development (Atzberger & Paetzold, 2019).

Significant challenges in hardware development that are unique compared to software development are related to the constraints of physicality (Atzberger & Paetzold, 2019). Hardware products being mechatronic or cyber-physical, containing mechanical, electrical and software components also bring its own challenges to iterative APM (Atzberger et al., 2019). Atzberger and Paetzold (2019) write that the production and testing of hardware prototypes essentially demand more time and resources than software iterations, making it difficult to deliver shippable increments within short development cycles. Long lead times for parts, especially from external suppliers, further restrict rapid iterations (Atzberger & Paetzold, 2019). Additionally, extended timelines for producing custom tools and completing certifications, particularly in safety-critical areas, create significant external dependencies (Atzberger & Paetzold, 2019). Atzberger and Paetzold (2019) add that the specialisation required in mechatronic development also constraints team availability, and coordinating across expertise domains and interfaces add further layers of complexity to agile practices. Moreover, synchronization between hardware, software and electronic further complicates iteration cycles and frequent feedback loops are harder to maintain due to stakeholder expectations of more complete and functional prototypes (Atzberger & Paetzold, 2019). On the other hand, Ciric et al. (2019) note that many of the challenges faced in hardware development, such as alignment among stakeholders, prioritization, insufficient time for testing and long feedback loops are also faced in software development. Many challenges may be rooted in the incompatibility of agile practices with organizational processes and functions, which Ciric et al. (2019) have also identified as one of the core challenges. Nevertheless, it's evident that the tangible nature of physical products brings many unique challenges to applying APM in hardware development.

These physical limitations directly affect APM practices such as continuous integration, automated testing and maintaining stable interfaces between teams (Atzberger & Paetzold, 2019). Nevertheless, evidence suggests that Agile can still bring benefits in hardware contexts. Gustavsson and Rönnlund (2013) write that even in hardware development, agile principles such as iterative work, simulation-based testing and team-level

flexibility can enable responsiveness to change and decrease development time. Simulating environments and advanced testing techniques help hardware teams iterate effectively despite physical constraints, enabling some of the core Agile benefits like improved adaptability and prioritization of the most valuable work (Gustavsson & Rönnlund, 2013; Weichbroth, 2022). Gustavsson and Rönnlun (2013) note that unlike software which can be patched or upgraded post-deployment, hardware often requires a redesign or a completely new development cycle for changes. The teams from Gustavsson and Rönnlund's (2019) study mitigated these kinds of legacy issues by defining project boundaries to end development at production handover and thereby avoiding costly post-production upgrades (Gustavsson & Rönnlund, 2013).

Weichbroth (2022) presents three different Agile-oriented frameworks that are particularly suitable for hardware development: Scaled Agile Framework (SAFe), Scrum for hardware and Modified Agile for Hardware Development (MAHD). Additionally, Weichbroth (2022) presents hybrid Stage-Gate framework but this and other hybrid frameworks are later covered in the chapter of hybrid project management.

The Scaled Agile Framework (SAFe) was developed for scaling agile practices across large enterprises, particularly in environments that require coordination among multiple teams and stakeholders (Ciancarini et al., 2022; Putta et al., 2018). Because of its suitability for implementing agile practices in large multi-team enterprises, SAFe has gained widespread popularity and is one of the most popular multi-team Agile methods (Ciancarini et al., 2022; Sreenivasan & Kothandaraman, 2019). SAFe is built on combining multiple agile methodologies: Scrum, Extreme Programming, Kanban and lean. Furthermore, SAFe expands these practices to work across four different levels of an organization: team, program, value stream and portfolio (Alqudah & Razali, 2016; Putta et al., 2018). All levels are tied together and each has their own roles, responsibilities, activities and structures (Alqudah & Razali, 2016; Ciancarini et al., 2022). Team level consists of agile teams that work as a typical scrum team (Alqudah & Razali, 2016; Putta et al., 2018). Program level consists of multiple agile teams that are dedicated to the program and

create value to the program (Alqudah & Razali, 2016). At the program level, SAFe introduces Agile Release Trains (ARTs) which synchronize multiple agile teams through fixed-length iterations to develop large-scale systems (Alqudah & Razali, 2016: Putta et al., 2018). ARTs are followed by HIP (hardening, innovation, planning) iterations to produce potentially shippable increments or program increments (Alqudah & Razali, 2016; Putta et al., 2018). Alqudah and Razali (2016) write that a value stream refers to a continuous set of activities involved in defining, developing and delivering systems that provide ongoing value to the business, customer or end user. Value streams are realized in SAFe through ARTs at the program level while investment themes indicate how the portfolio allocates funding to these ARTs in alignment with strategic objectives (Alqudah & Razali, 2016). These themes act as a funnel by filtering and translating strategic goals into business or technical epics within the portfolio backlog (Alqudah & Razali, 2016).

Ciancarini et al. (2022) write that SAFe is based on four core values: business alignment, build-in quality, transparency and program execution. These core values are cited as contributing factors to enhanced productivity, quality, accelerated time-to-market, transparency and improved team collaboration (Ciancarini et al., 2022; Putta et al., 2018). Weichbroth (2022) writes that in the context of hardware development, SAFe offers six tailored principles to address the hardware-specific constraints: organizing around value, assuming variability and preserving options, designing for change, building incrementally and integrating frequently, performing work in small batches, and establishing continuous integration for hardware. These principles aim to guide hardware teams in creating flexible and iterative development processes suited to the complexities of physical product development. However, while the structured and prescriptive nature of SAFe is beneficial for guiding large-scale system development, it has also drawn criticism. Its complexity and resource-intensive implementation present challenges such as resistance to change, difficulty in aligning distributed teams, test automation challenges and concerns about reduced agility (Ciancarini et al., 2022; Putta et al., 2018). Furthermore, some argue that the rigidity of SAFe may conflict with the foundational Agile value of adaptability (Ciancarini et al., 2022). Nevertheless, Ciancari et al. (2022) find that when SAFe is fully

supported by leadership and understood across the organization, it can facilitate systematic improvements in the earlier mentioned benefits as well as in strategic alignment and enterprise-wide agility.

Scrum for hardware is an adaptation of the traditional scrum framework tailored to meet the unique demands of hardware and system development (Weichbroth, 2022). According to Weichbroth (2022) it integrates modular design principles and practices from lean and extreme programming. The framework outlines eight key focus areas to support successful implementation to hardware development: managing uncertainty in hardware projects, using stubs and mock-ups, promoting modular designs, enabling continuous integration, designing clear interfaces, applying test- and data-driven developments, forming effective teams and ensuring a functional product is delivered at the end of each sprint (Weichbroth, 2022).

The third framework Weichbroth (2022) presents is modified Agile for hardware development (MAHD) which was developed by Simpson and Hinkle. MAHD framework was particularly developed for hardware that combines electronics, mechanical components and software (Simpson & Sheth, 2025). MAHD is based on four foundational principles: short development cycles for continuous learning and adaption to change, autonomous teams, validating work at the end of each development cycle, and applying rapid prototyping strategies. Simultaneously, MAHD addresses the physical product requirements by allowing design freezes but also flexibility to change, quality planning, managing supply chain, defining product attributes for electrical and mechanical components, developing documentation, developing flexible prototyping and product validation strategies, and sharing resources and external partners (Weichbroth, 2022).

MAHD includes the familiar agile practices, such as iterative development cycles and prioritized product backlog, but additionally introduces four key differences which take the unique aspect of hardware into account (Simpson & Sheth, 2025). The first difference comes on initiating a project which requires additional upfront planning to create

structure and initiate the right direction compared to software development, where defining user stories is enough to start the project (Simpson & Sheth, 2025). Planning is done fast but comprehensively enough, including five on-ramp steps of refining the project: user stories, product attributes, focus matrix, IPAC iteration plans and task backlog. Simpson and Sheth (2025) highlight that although these on-ramp steps might resemble early waterfall activities, they are fundamentally different, focusing on collaboration and quick alignment on critical success factors to provide just enough detail to move forward confidentially. The second difference, according to Simpson and Sheth (2025), is that instead of one iterative development cycle, MAHD includes two levels of development cycles to align and manage the various disciplines required in hardware development. Additionally, to general sprints, there are also IPAC (integration, prototype, alignment and customer) iterations on product level which last between 1 to 6 sprints (Simpson & Sheth, 2025). Simpson and Sheth (2025) explain that these IPAC iterations focus on delivering system-level outcomes, ensuring integration across disciplines and subsystems, validating the solutions and minimizing risk. The third difference is that hardware teams require a flexible backlog that allows them to organize tasks by discipline and subsystem, follow iteration milestones and manage additional requirements unique to hardware projects compared to software development (Simpson & Sheth, 2025). The fourth difference comes from the team roles defined in MAHD which differ a little from generalized scrum roles. Simpson and Sheth (2025) write additional to product owner, MAHD also includes product manager who acts as the ultimate decision-maker, making broader and critical decisions on scope, schedule and resources, while a supporting internal product owner keeps teams focused on customer value. Replacing the scrum master, MAHD has an agile project leader who manages cross-disciplinary coordination, removes obstacles and lead IPAC iteration planning with a stronger focus on facilitation and stakeholder alignment (Simpson & Sheth, 2025). MAHD also includes discipline-specific and system-level technical leaders who define iteration goals, lead prototyping and manage their areas of the backlog (Simpson & Sheth, 2025). Simpson & Sheth (2025) also explain that while scrum teams are typically cross-functional and dedicated to one project at a time, MAHD team

members include full- and part-time contributors across functions who commit to deliverables despite often being shared across multiple projects.

2.1.2 Strengths and Weaknesses of Agile and Scrum

APM has been widely recognized across industries for its diverse strengths and benefits. These benefits can be divided into three primary categories: business, product and process, and team. Additionally, some benefits of scaling agile are introduced. Some of the benefits can also be included in multiple categories.

One of the significant benefits of Agile is the increased focus on achieving meaningful business outcomes rather than accelerating delivery (Cobb, 2023). According to Cobb (2023), Agile emphasizes delivering value through adaptive and iterative methods that prioritize successful outcomes over speed alone. This approach enables organizations to align projects with strategic goals and customer expectations, which in turn improves stakeholder satisfaction (Ciric et al., 2019; Gemino et al., 2021). Gemino et al. (2021) found that Agile and hybrid projects significantly outperformed traditional approaches on stakeholder success measures although all three approaches, Agile, hybrid and traditional, showed similar results on time, scope and budget. This supports the argument that the value-oriented principles of Agile strengthens stakeholder success without compromising the triple constraint success metrics.

Agile also reduces time-to-market by simplifying requirements definition and delivering incremental value more efficiently (Cobb, 2023). It allows for quicker reaction to changing market conditions and customer demands, thus improving responsiveness and performance results (Santos & De Carvalho, 2022; Thesing et al., 2021). Furthermore, Ciric et al. (2019) writes that agile improves the organization's ability to manage changing priorities. This can improve financial outcomes by reducing effort and increasing development speed through lean processes (Karrenbauer et al., 2019). Additionally, Agile facilitates the transfer of knowledge and know-how, enabling team to better understand and respond to evolving customer needs (Ciric et al., 2018). The integration of lead users

and early market alignment ensures that new product development is oriented toward future customer needs, which offers a strategic business advantage and strengthens stakeholders and customer commitment (Ciric et al., 2018).

Agile practices also result in significant improvements in product development and project execution. Agile approach supports excellent product outcomes by embedding quality assurance into each step of the development cycle (Cobb, 2013; Karrenbauer et al., 2019; Thesing et al., 2021). Cobb (2023) explains that in contrast to the Waterfall model, where quality is often a separated step, Agile assigns quality responsibility to the entire team, promoting a shared commitments to building quality. Frequent feedback loops and short iterations support early error detection and continuous refinement (Thesing et al., 2021; Karrenbauer et al., 2019). This iterative structure provides flexibility to innovation and to manage uncertainty effectively (Cobb, 2023; Ciric et al., 2018; Thesing et al., 2021). Teams are encouraged to develop exploratory prototypes and conduct incremental testing to experiment with partial solutions, clarify requirements and refine the product over time (Cobb, 2023; Ciric et al., 2018; Thesing et al., 2021). According to Karrenbauer et al. (2019), the flexibility and adaptability improved by this iterative structure is particularly beneficial in complex or regulated environments where requirements can change quickly and issues such as compatibility with modules and integration with hardware exists. Furthermore, Santos and De Carvalho (2022) write that Agile improves the management of evolving requirements through active stakeholder engagement, resulting in products that align closely with user needs. Karrenbauer et al. (2019) also add that Agile improves usability through simplified documentation and transparency. This transparency is enabled through continuous feedback and ongoing comparison between planned and actual progress (Karrenbauer et al., 2019). Additionally, the use of lean frameworks like Scrum allows teams to remain closely aligned with real-world conditions when frequent adjustments are made during each iteration to better reflect evolving project realities (Karrenbauer et al., 2019). With these characteristics, APM also supports risk mitigation by allowing for early identification of deficiencies, thus improving failure management processes (Santos & De Carvalho, 2022; Thesing et al., 2021).

Agile strengthens collaborative and dynamic work environments that improve team functioning and motivation (Karrenbauer et al., n.d.; Santos & De Carvalho, 2022). By promoting frequent feedback and collaboration, Agile helps build a culture of shared responsibility and ownership and mutual respect which also improves project visibility (Ciric et al., 2018; Karrenbauer et al., n.d.; Santos & De Carvalho, 2022). This environment further contributes to increased morale and greater job satisfaction (Cobb, 2023; Karrenbauer et al., 2019). The iterative structure of Agile also strengthens team motivation by reinforcing transparency and adaptability (Karrenbauer et al., n.d.; Santos & De Carvalho, 2022). Teams continuously learn and improve their processes, evolving toward more effective and efficient practices (Karrenbauer et al., n.d.; Santos & De Carvalho, 2022). According to Thesing et al. (2021), these characteristics are especially valuable during early development phases where speed and flexibility are essential.

Despite the significant challenges of adopting large-scale APM, Santos and De Carvalho (2021) highlighted some benefits that emerge when agile principles are effectively scaled. One of the primary benefits is increased flexibility within mature organizations, showing the suitability of agile also for large and complex projects with long life cycles (Santos & De Carvalho, 2021). Improved collaboration during planning events leads to stronger alignment with business requirements and facilitates shared ownership and responsibility across organizational units (Santos & De Carvalho, 2021). The emphasis on continuous feedback and client engagement contributes to higher levels of trust, transparency, and responsiveness in product development (Santos & De Carvalho, 2021). Santos & De Carvalho (2021) also write that the identification of critical paths and inter-team dependencies enables more effective prioritization of requirements, minimizing resource waste and improving traceability, testing capabilities, and product quality. To achieve these benefits, it’s critical that the organization has strategic alignment, strong support and shared understanding of the benefits of adopting large-scale APM (Santos & De Carvalho, 2021).

Table 1. Strengths of APM

Strengths	Sources
Business: • Focus on achieving meaningful business outcomes • Alignment of projects with strategic goals and customer expectations • Improved stakeholder satisfaction and success • Reduced time-to-market through incremental delivery • Improved responsiveness to changing market conditions • Ability to manage changing priorities • Increased financial outcomes through lean processes • Knowledge transfer and customer understanding • Early market alignment and customer commitment	Cobb (2023); Ciric et al. (2018, 2019); Gemino et al. (2021); Karrenbauer et al. (2019); Santos & De Carvalho (2022); Thesing et al. (2021)
Product and process: • Embedded quality assurance throughout development • Early error detection and continuous improvement • Support for flexibility and innovation through iterations • Active stakeholder engagement ensures alignment with user needs • Improved usability, transparency, and documentation simplification • Improved adaptability in complex environments • Improved risk mitigation through early deficiency identification	Cobb (2023); Ciric et al. (2018); Karrenbauer et al. (2019); Thesing et al. (2021); Santos & De Carvalho (2022)
Team: • Strengthened collaboration and motivation • Culture of shared responsibility and transparency • Increased morale, mutual respect, and job satisfaction • Continuous learning and process improvement • Higher adaptability and visibility within projects	Cobb (2023); Ciric et al. (2018); Karrenbauer et al. (2019, n.d.); Santos & De Carvalho (2022); Thesing et al. (2021)

Scaling Agile: • Increased flexibility in large and complex projects • Improved collaboration and shared ownership across units • Increased trust, transparency, and responsiveness • Improved prioritization and resource utilization • Improved product quality and traceability through inter-team coordination	Santos & De Carvalho (2021)

Although APM has proven effective across various industries due to its adaptive and iterative nature, its implementation still faces several challenges. The identified challenges can be divided into following categories: organizational challenges, managerial challenges, agile method barriers, customer challenges, team challenges, product and process challenges and scaling challenges.

Organizational resistance to agile adoption remains a common challenge. Ciric et al. (2019) identify the incompatibility of agile methods with existing organizational processes and functions as a common challenge, both within and beyond software development. Ciric et al. (2019) emphasize that in many cases agile methods conflict with traditional business structures that rely on rigid planning cycles, linear reporting structures and detailed documentation. This misalignment limits the flexibility and iterative improvements that are central in agile methods (Ciric et al., 2019). Additionally, the lack of formal project management strategy, guidelines and standardized processes complicate the integration of Agile (Ciric et al., 2019). Atzberger & Paetzold (2019) also note the challenge of establishing an agile mindset among individuals. Teams that are used to traditional, plan-driven approaches often struggle to rethink their work through agile principles (Atzberger & Paetzold, 2019). The transition requires not only a shift in working methods but also a deeper cultural change in how individuals approach teamwork and accountability (Atzberger & Paetzold, 2019). Moreover, Atzberger & Paetzold (2019) note that although Agile promotes transparency in many ways, if the environment and

culture is not safe and truly support the transparency, individuals may withhold from openly bringing up challenges and mistakes. Ultimately, this can lead to frustration, disengagement and resistance to Agile adoption, undermining its effectiveness (Atzberger & Paetzold, 2019).

Managerial challenges significantly affect the successful adoption and operation of agile methods. A common challenge is the difficulty in achieving senior management sponsorship. Hoda and Murugesan (2016) found that reluctance to support agile practices often comes from senior management's focus on final deliverables and discomfort with changing established development processes. Moreover, senior management may be hesitant to fully sponsor Agile transformation because of discomfort with iterative delivery models and reduced control over detailed planning (Hoda & Murugesan, 2016). Even when agile methods were accepted by senior management, requirements for extensive documentation were still required, conflicting with the agile practices of only involving minimal documentation (Hoda & Murugesan, 2016). According to Hoda & Murugesan (2016), another managerial gap involves limited investment in team development. They note that some managers are unwilling to support professional training or certification for agile teams, perceiving it as unnecessary or low priority. This resistance to skill-building compromises the team's ability to integrate and effectively apply agile principles (Hoda & Murugesan, 2016). Furthermore, managerial misunderstanding or misapplication of agile roles and responsibilities creates conflict. Managers may unintentionally micromanage teams or continue enforcing hierarchical control models, limiting self-organization and cross-functional collaboration (Hoda & Murugesan, 2016). Schwaber and Sutherland (2020) emphasize the importance of servant leadership within Agile settings, where managers facilitate rather than direct team efforts.

Agile method and its characteristics also have practical challenges. Scope management is a central challenge in Agile projects since the foundational aspect of flexibility and adaptability also increases the risk of uncontrolled scope creep if changes are introduced without proper reprioritization of the backlog (Sithambaram et al., 2021). Sithambaram

et al. (2021) write that although Agile approaches avoid extensive upfront planning, it should still be done as much ahead as possible considering all available information and predictable elements. In addition, risk management shouldn't be neglected because risk identification and mitigation are also important in Agile to prevent being surprised by a risk that could have been foreseen (Sithambaram et al., 2021). In addition, Hoda and Murugesan (2016) identified the pressure to manage interruptions and urgent requests in the middle of a sprint as a challenge. These disruptions risk sprint integrity and reduce team focus and productivity (Ciric et al., 2019; Hoda & Murugesan, 2016). Although Scrum promotes maintaining a stable work environment throughout the sprint, practical realities often force teams to respond to unplanned tasks which reduces velocity and delivery predictability. Another challenge occurs in the difficulty of workload estimation during collaborative sprint planning (Hoda & Murugesan, 2016). Hoda and Murugesan (2016) explain that while Agile encourages team-based estimation practices, differences in experience levels as well as understanding the product and unclear requirements often lead to unreliable estimates, impacting sprint outcomes and stakeholder expectation.

Customer-related challenges mainly revolve around communication and expectation management. Hoda and Murugesan (2016) emphasize the difficulty in gathering clear and timely requirements from customers. While customers usually are able to give high-level requirements, gathering detailed requirements and acceptance criteria, which are critical for accurate development and testing, turns out to be a challenge (Hoda and Murugesan, 2016). Additionally, Ciric et al. (2019) write that beyond software development, visibility into client value and predictability of delivered business value are significant concerns. These challenges indicate a disconnect between agile teams and customer expectations, which might affect stakeholder satisfaction.

Team dynamics pose several challenges in Agile environments since teams are expected to be self-organized and autonomous. Hoda and Murugesan (2016) write that maintaining autonomy, particularly in newly formed and transitioning teams, requires a cultural shift that is difficult to achieve. Some team members might also resist or struggle

assuming responsibility or being self-direct (Hoda & Murugesan, 2016). Additionally, the lack of clarity in task-level requirements, including missing acceptance criteria and high task dependency, further hinders effective team performance (Hoda & Murugesan, 2016). Hoda and Murugesan (2016) state that if the task-level issues are left unresolved, it leads to confusion, duplicated efforts and reduced velocity.

Generally, from the challenges collected by Santos and De Carvalho (2021) the ones related to product and process category are not cited as many times as the challenges in other categories. This implies product and process challenges, such as quality and progress measurement, may not be as significant barriers to Agile adoption than challenges identified in other categories. However, project size is the one challenges that rises from the product and process category (Santos & De Carvalho, 2021). Santos and De Carvalho (2021) explain that project size is a significant challenge because agile practices need to be implemented differently in large projects when project complexity increases and the complexity further decreases the accuracy of task effort estimates. Moreover, adopting APM to large-scale project brings its own additional challenges, involving managing technical complexity but also addressing a range of organizational, cultural and structural factors. Santos and De Carvalho (2021) have identified geographic distribution, regulatory compliance, organizational and cultural complexities and corporate strategy issues as critical factors related to large-scale projects where agile teams are scaled. These factors introduce significant coordination demands and often reduce the autonomy of self-managed agile teams (Santos & De Carvalho, 2021). Additionally, leadership and managerial engagement play a critical role in successful scaling of Agile. Santos & De Carvalho (2021) write that leadership must actively shape strategy, provide cross-functional support and facilitate structural changes that reinforce agile values. Traditional project management metrics, such as the project triple constraints of scope, cost and time, must be adjusted to accommodate the iterative and adaptive characteristics of agile (Santos & De Carvalho, 2021). Furthermore, APM's focus on speed and continuous delivery instead of documentation and specifications can potentially weaken auditability and compliance (Santos & De Carvalho, 2021). This can be particularly challenging when Agile methods

are used in industries that require strict certifications, such as ISO and CMMI, which typically demand extensive documentation and can also be time-consuming (Atzberger & Paetzold, 2019; Santos & De Carvalho, 2021).

Table 2. Challenges of APM

Challenges	**Sources**
Organizational: • Incompatibility with existing organizational processes • Resistance to agile adoption • Lack of formal strategy and standardized processes • Difficulty establishing an agile mindset • Unsafe culture discouraging transparency	Ciric et al. (2019); Atzberger & Paetzold (2019)
Managerial: • Lack of senior management sponsorship • Discomfort with iterative delivery and reduced control • Excessive documentation requirements • Limited investment in team development • Misunderstanding of agile roles and servant leadership gaps	Hoda & Murugesan (2016); Schwaber & Sutherland (2020)
Method barriers: • Scope creep • Sprint interruptions and unplanned work • Difficulty with work load estimation accuracy • Reduced predictability of delivery • Pressure to balance flexibility and structure	Hoda & Murugesan (2016); Ciric et al. (2019); Sithambaram et al., 2021
Customer: • Difficulty gathering detailed requirements • Misaligned expectations and satisfaction gaps	Hoda & Murugesan (2016); Ciric et al. (2019)

• Limited visibility into client value and business predictability	
Team: • Difficulty achieving self-organizing teams • Resistance to accountability and autonomy • Lack of clarity in task requirements • High task dependency and confusion • Reduced velocity and duplicated effort	Hoda & Murugesan (2016)
Product and process • Difficulty measuring quality and progress • Project size increases complexity and reduces estimate accuracy • Scope management issues (scope creep, unclear definition) • Cost management and resourcing difficulties • Lack of upfront planning and risk management	Santos & De Carvalho (2021); Sithambaram et al. (2021)
Scaling Agile: • Complexity in coordination and inter-team dependencies • Geographic, regulatory, and cultural barriers • Compliance and documentation constraints (ISO, CMMI) • Reduced team autonomy in scaled contexts • Leadership engagement and strategic alignment gaps • Misfit of traditional metrics (scope, cost, time) • Limited auditability and traceability	Santos & De Carvalho (2021); Atzberger & Paetzold (2019)

2.2 Traditional project management & Waterfall model

According to Wysocki (2019), TPM formed the foundation for modern project management. TPM, often referred as waterfall method, is a linear project management

methodology that assumes that the circumstances of the project are predictable and clear (Gemino et al., 2021; Špundak, 2014; Zasa et al., 2021). It contains distinct goals for each phase of development (Breyter, 2023). Each phase of the development process is only done once in linear order, thus the entire product is tested and delivered all at once at the end of a project (Cobb, 2023; Wells, 2012). This is why TPM methodologies are particularly suitable for projects that have been executed before or have low uncertainty (Cobb, 2023). Detailed project plan is defined at the beginning of the project based on customer requirements that are assumed to be clear and stable without any need for changes during the project (Ciric et al., 2019). Because of the assumed predictability, TPM has a strong focus on planning and control, and the goal is to execute the plan efficiently to complete the project within the planned triple constraints of scope, time and cost (Ciric et al., 2019; Gemino et al., 2021). Over time, multiple frameworks have been developed for executing TPM, most notable the Stage-Gate model for NPD, providing a systematic way to plan, execute and control projects.

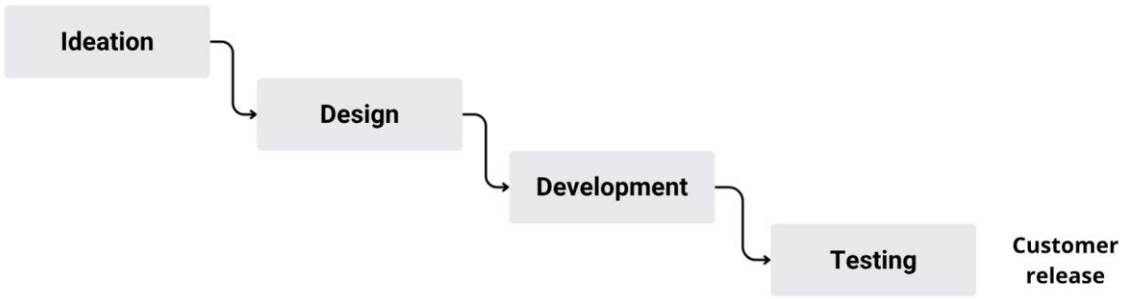

Figure 3. Waterfall model (adapted from Breyter, 2023)

2.2.1 Stage-Gate model

Stage-Gate moddel is a cross-functional model that breaks down a NPD project into sequential stages from idea generation to the launch of the product, emphasizing pre-development tasks and planning (Cooper, 2016). It also acts as an action guide, providing specific best practices for each stage (Cooper, 2016). Furthermore, the model aims to ensure process quality in innovation development by providing structured guidance and discipline at each step (Reiff & Schlegel, 2022). It was developed to address the

challenges of managing unstructured development projects by providing structure, control, excessive front-end planning and work progress monitoring (Bianchi et al., 2020). This linear and plan-driven approach, requiring early decisions and adherence to predefined plans, has been a central factor in its widespread adoption across industries for decades (Reiff & Schlegel, 2022; Zasa et al., 2021). According to Reiff & Schlegel (2022), a foundational principle of the original Stage-Gate model is the early fact-based and clear definition of products to improve project outcomes. Detailed product specifications and a project plan are defined early in the process, as well as design freeze, to avoid delays in the product design and implementation stages (Bianchi et al., 2020; Cooper & Sommer, 2018). Bianchi et al. (2020) and Cooper and Sommer (2018) note that once design freeze is done, any later modifications are seen as troublesome since Stage-Gate doesn't support reversible iterations and later modifications can be costly.

There are multiple variations of the Stage-Gate model but typically the model contains six stages, including idea screening, with defined tasks and deliverables to be fulfilled (Cooper, 2023; Cooper & Sommer, 2018; Sommer et al., 2015). Furthermore, the model includes five gates between the stages where go/kill investment decisions are made, creating a funnel model (Cooper, 2023; Cooper & Sommer, 2018; Sommer et al., 2015). The model is presented in the Figure 4. Cooper (2016) describes that the go/kill decisions at each gate guide resources to the most potential and beneficial projects as resources are allocated for projects that get a go-decision for the next stage.

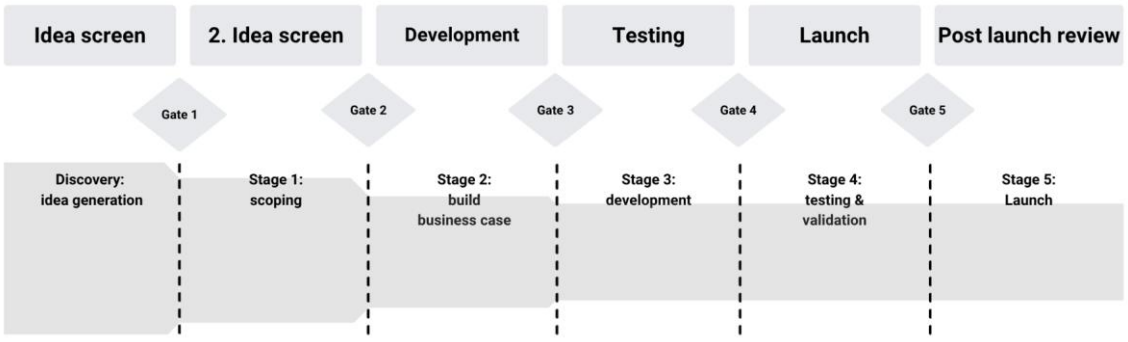

Figure 4. Stage-Gate model (adapted from Cooper, 2023)

The first stage, sometimes referred as stage 0, is dedicated for discovering and generating new potential product ideas (Cooper, 2023). During this stage technical research is done and for potential new product ideas a one-pager is done for describing the idea (Cooper, 2023). Cooper (2023) highlights that stage 0 for ideation is not officially part of the project, but because it's considered essential by many organizations, it is also important to handle formally, building a defined and proactive system for idea generation. Stage 0 is followed by gate 1 in which the product idea is screened based on qualitative criteria such as project feasibility, strategic alignment, market opportunity and ability to leverage the organization's resources (Cooper, 2023). Cooper (2023) highlights that financial criteria is not typically done during gate 1 because there's not enough information yet. The go/kill investment decision to start the project is made based on the screening and resources are committed to the project (Cooper, 2023). Stage 1 is typically relevantly short stage dedicated for scoping and determining the technical and marketplace merits of the project (Cooper, 2023).

In gate 2, a second more rigorous screening is done, mainly re-evaluating the same criteria of gate 1 with the new information gathered from stage 1 (Cooper, 2023). Additionally, financial criterion is lightly assessed before making the go decision to move to stage 2 (Cooper, 2023). Cooper (2023) describes stage 2 as a "make or break" stage where most new product development failures happen. In stage 2, a cross-functional team conducts a detailed investigation about the product and a business case is constructed (Cooper, 2023). This includes for example a market analysis, customer research, competitive analysis, concept testing, detailed technical assessment and detailed business and financial analysis (Cooper, 2023).

Project definition, justification and detailed plan are made for moving to gate 3, which the last gate before development stage with substantial financial commitments (Cooper, 2023). Cooper (2023) explains that all tasks from stage 2 are reviewed and financial analysis is particularly carefully examined to make sure results are positive to pass the gate to stage 3. In stage 3, the actual development of the product is started with the

implementation of the project plan (Cooper, 2023). Cooper (2023) writes that for large projects, multiple milestones and reviews can be included in this stage as check points, providing more control for project management. Gate 4 acts as a post-development review to check on the process to confirm the development fulfils quality aspects and product is consistent with the original definition (Cooper, 2023). Financial analysis is also reviewed and updated based on current data (Cooper, 2023). Furthermore, test and validation plans are made and approved for moving to stage 4 for immediate implementation (Cooper, 2023). In stage 4, the viability of the project is tested and validated though multiple aspects: the product, production process, financial and customer acceptance (Cooper, 2023). Cooper (2023) writes that this includes tasks such as products tests, user trials, pilot production and operations and trial sells, and furthermore the business and financial analyses are revised. Cooper (2023) adds that if this stage is not successful, it is possible to go back to stage 3 to continue development.

A successful stage 4 moves to the gate 5 which is the final gate before full production and market launch (Cooper, 2023). According to Cooper (2023), this gate mainly acts as a check for readiness to move to full production based on reviewing the results of stage 4. Passing gate 5 is followed by final stage 5 of product launch where the product is moved from the project to operations (Cooper, 2023). The project is officially terminated when the new product is fully commercialized, and a post-launch review is conducted reviewing the performance of the project (Cooper, 2023).

2.2.2 Strengths and weaknesses

A core benefit of TPM, specifically the widely adopted Stage-gate mode, is its emphasis on rigorous planning and early decision-making. Cooper (2023) writes that with each step of the Stage-Gate model more information is gathered, and uncertainties are reduced which is why the model is also presented as a risk mitigating model. Rigorous upfront planning and freezing of the product design foster stability, discipline and compliance which can lower development costs, enable timely project completion and improve product quality (Bianchi et al., 2020). However, Bianchi et al. (2020) note that these

benefits come from the Stage-Gate model's assumption that problems and solutions can be anticipated in advantage and risk can be managed proactively with careful planning. Another advantage is efficiency in resource allocation since in each stage cross-functional teams execute tasks both concurrently and in parallel, and with each gate passed more resources are allocated for the project (Cooper, 2023). Furthermore, TPM methodology provide fixed roles and responsibilities, stable processes, systematic documentation, reliable estimation of time and budget and measurable progress through milestones (Thesing et al., 2021). Additionally, because customer feedback is typically collected only at the end of the process teams can avoid lengthy and costly prototyping stages, instead focusing on refining the most promising design for launch (Bianchi et al., 2020). These characteristics make TPM approach suitable where predictability, accountability and structured governance are essential. From a wider perspective, TPM is also valued for its ability to provide managerial control, organizational structure and transparency. Milestone reviews and gate approvals in the Stage-Gate model allow management to monitor progress closely and make informed go/kil decisions, reducing the risk of investing in unfavourable projects (Cooper & Sommer, 2018).

Table 3. Strengths of TPM

Strengths	**Sources**
Business: • Front-end planning and early decision-making • Reduced uncertainty and improved risk mitigation • Stability, discipline and compliance lower costs • Improved quality	Bianchi et al., 2020; Cooper, 2023
Managerial:	Cooper, 2023; Cooper & Sommer, 2018; Thesing et al., 2021

• Efficient resource allocation through gate-based decision-making • Improved process monitoring • Structured governance and transparency through documentation and milestones • Clear fixed roles	
Product and process: • Concurrent and parallel task execution improves process efficiency • Avoidance of lengthy prototyping stages saves time and focuses resources	Bianchi et al., 2020; Cooper, 2023

Despite these advantages, TPM methodologies face significant limitations when applied to uncertain or rapidly changing environments (Ciric et al., 2019; Cooper & Sommer, 2018). One of the central criticisms is their lack of adaptability, as they are often considered too rigid and linear to effectively respond to evolving market and customer needs, which are common in modern projects (Cooper & Sommer, 2018; Reiff & Schlegel, 2022). Bianchi et al. (2020) and Thesing et al. (2021) note that it is difficult to fully gather user needs or design alternatives during the early planning stages, particularly in NPD projects. One main reason for this is that customers often struggle to define all the requirements at the early stages (Ciric et al., 2019). The linear process of Stage-Gate model means that key decisions are made before all essential information is available, increasing the possibility of errors (Bianchi et al., 2020; Thesing et al., 2021). Late customer feedback and the inflexibility to adapt in later stages of the project increase the risk of discovering misalignments with customer and market needs only after significant resources have been invested (Bianchi et al., 2020). These late-stage corrections can cause resource-intensive rework and costly delays (Bianchi et al., 2020). Gemino et al. (2021)

also note that the likelihood of stakeholder dissatisfaction increases with late-stage misalignments and corrections. Additionally, according to Bianchi et al. (2020), proactive risk management often leads to over-specification of products, resulting in unnecessary complexity or gold plating where unrequested features are delivered. In the end, the heavy reliance on detailed upfront planning can also mean that large portions of the project plan must be discarded when too many changes become necessary (Maier & Emmerich, 2022).

Table 4. Challenges of TPM

Challenges	**Sources**
Business: • Lack of adaptability and rigidity in responding to changing customer and market needs	Cooper & Sommer, 2018; Reiff & Schlegel, 2022
Managerial: • Decision-making based on incomplete early-stage information increases error	Bianchi et al., 2020; Thesing et al., 2021
Product and process: • Late customer feedback can cause misalignment and costly rework • over-specification and gold-plating due to proactive risk management • Discarding of plans when excessive changes occur undermines efficiency	Bianchi et al., 2020; Gemino et al., 2021; Maier & Emmerich, 2022

2.3 Hybrid project management

HPM refers to the combination of traditional plan-driven methodologies and adaptive Agile methodologies (Mirzaei et al., 2025; Reiff & Schlegel, 2022; Székely et al., 2025). Reiff and Schlegel (2022) have identified two primary definitions of HPM. One describes HPM as a combination of TPM and APM by applying TPM practices at the decision-making level and agile practices at the operational level (Reiff & Schlegel, 2022). The other definition describes HPM as a result from integrating APM practices into already existing TPM methodologies (Reiff & Schlegel, 2022). In both cases the goal is to combine the strengths of APM and TPM methodologies while avoiding their limitations (Reiff & Schlegel, 2022; Székely et al., 2025). Several studies have found that most Agile implementations are in fact using hybrid methods rather than a pure Agile approach (Bianchi et al., 2022; Edwards et al., 2019; Gemino et al., 2021; West et al., 2011). Reed et al. (2024) also found that despite many stating following only agile practices, predictive techniques and tools were still often used. One reason for this is that Agile adoption is often being practitioner-led, which leads to development teams using Agile but areas outside their control continue following more TPM methodologies (Gemino et al., 2021; West et al., 2011). Furthermore, this may suggest that organization's leadership and project managers must balance between predictability and adaptability and therefore struggle to follow pure agile practices (Reed et al., 2024; Sithambaram et al., 2021).

The growing relevance of HPM reflects the increasing complexity, uncertainty and continuous rapid change in modern project environments (Székely et al., 2025; Zasa et al., 2021). According to Zasa et al. (2021) traditional methods such as the Stage-Gate model have proven less effective when product development activities become unpredictive, involve frequent rework or require adaptation even in late stages of the project. Today's fast-changing conditions also challenge the feasibility of following the Stage-Gate model's linear step-by-step structure (Cooper, 2016; Zasa et al., 2021). Contrarily, relying solely on Agile methods can be challenging because they lack the structure, governance and documentation that especially large-scale organizations require (Sithambaram et al., 2021; Zasa et al., 2021). West et al. (2011) also note that industries with strict compliance

requirements require well established governance processes in NPD. For these reasons HPM has emerged with the aim of combining the strengths of both Agile and TPM methodologies, resulting in an adaptive and responsive project management methodology that simultaneously offers enough control and documentation (Cooper, 2016; Cooper & Sommer, 2018; Reiff & Schlegel, 2022; Sommer et al., 2015; Székely et al., 2025; Zasa et al., 2021). According to Reiff & Schlegel (2022), this balance is particularly valuable for organizations where innovation and adaptability must be maintained without undermining planning reliability. Székely et al. (2025) write that industries such as manufacturing, R&D, IT and SC management are particularly suitable for implementing HPM because these industries often have complex, dynamic and high-stakes projects where it's essential to balance flexibility and structure. Azenha et al. (2021) note that HPM is primarily used in projects with complex scope and high technical uncertainty where market demands are presumed to change, requiring adaptability and quick responsiveness. Additionally, NPD projects in non-software industries being less modular and requiring cross-functional collaboration between multiple teams make long-term planning essential and benefit from HPM approach (Zasa et al., 2021).

Multiple reasons to why particularly large organizations are more likely to opt toward HPM rather than pure Agile methodology have been identified. Zasa et al. (2021) and Reed et al. (2024) write that large organizations likely have established and strong TPM practices in use, which they are not willing to abandon completely. Gemino et al. (2021) further reason this by organizations wanting to maintain control over managing large and complex projects with pre-set budgets and time requirements as well as governance, compliance, safety and documentation requirements. With simultaneously needing to be more adaptive, the organizational realities drive the shift toward HPM where project teams are blending methodologies and different practices that are the most suitable for their environment and context (Gemino et al., 2021). As a result, agile practices are frequently incorporated into specific stages of the project lifecycle, most often during execution phases such as project planning and controlling where customer needs and priorities often change (Gemino et al., 2021; Sommer et al., 2015; Zasa et al., 2021). On the

other hand, TPM structures are maintained for strategic phases such as portfolio management, product vision, project oversight and coordination (Gemino et al., 2021; Sommer et al., 2015; Zasa et al., 2021).

Several different HPM models have been developed, each blending APM and TPM methodologies in different ways to suit unique project needs (Gemino et al., 2021; Reiff & Schlegel, 2022). Reiff and Schlegel (2022) introduce four different HPM models: Water-Scrum-Fall, Waterfall-Agile, Hybrid V-model and Agile-Stage-Gate. Water-Scrum-Fall model applies the Waterfall model at the start of the project to bring structure to creating a project plan and initial documentation, and at the end of the project to ensure quality control and delivering the solution (Reiff & Schlegel, 2022). Agile and Scrum framework are applied for the development phase which is done in iterative development cycles (Reiff & Schlegel, 2022). According to Reiff and Schlegel (2022), the Water-Scrum-Fall model is suitable particularly for organizations that have been using TPM methodologies and want to move towards more agile way of project execution. Similarly to the Water-Scrum-Fall model, the Agile-Waterfall model integrates Agile into the Waterfall model in the development phase but also the final phase of the project is executed within Agile practices (Reiff & Schlegel, 2022). Reiff and Schlegel (2022) note that the Waterfall-Agile model doesn't clearly specify where, when or how traditional project planning transitions into Agile implementation but this can be decided depending on the project. Both models minimize initial doubts of the project by utilizing TPM approach in initial project planning while facilitating early and continuous delivery and feedback to meet customer requirements and minimizing rework and rescheduling (Reiff & Schlegel, 2022). Hybrid V-model uses traditional approach at the beginning and at the end of the project and Agile approach in development, implementation and testing phases, similarly to Water-Scrum-Model (Reiff & Schlegel, 2022). Hybrid V-model combines the verification and validation logic of the V-model, common in systems engineering and regulated industries, with the iterative development cycles of Agile, thereby ensuring compliance and quality assurance while enabling incremental delivery (Reiff & Schlegel, 2022; Sarhadi et al., 2022).

Among these, the Agile-Stage-Gate model, also known as Scrum-Stage-Gate, has gained particular attention in both research and practice (Figure 5). Agile-Stage-Gate model merges the iterative and customer-focused nature of Agile with the structured governance of the Stage-Gate model (Cooper, 2016; Sommer et al., 2015; Zasa et al., 2021). Cooper (2016) writes that the Stage-Gate model is used in macroplanning and Agile is utilized in microplanning. The Stage-Gate model is used for strategic go/kill decisions for portfolio management and resource allocation (Gemino et al., 2021; Sommer et al., 2015; Zasa et al., 2021) Agile iterations, or sprints in Scrum, are integrated within the execution phases of the Stage-Gate process, enabling rapid prototyping, continuous feedback and flexibility while still maintaining formal gates for decision-making, resource allocation and risk assessment (Cooper, 2016; Sommer et al., 2015). Cooper (2016) writes that APM practices often take place in the development and testing stages but more experienced organizations are integrating Agile-Stage-Gate across the entire process of NPD. By combining the adaptability of Agile with the structure of Stage-Gate model, Agile-Stage-Gate has proven effective particularly for managing new product portfolios by supporting iterative experimentation without sacrificing strategic oversights (Sommer et al., 2015; Zasa et al., 2021). Sommer et al. (2015) proved in their study that implementing Agile-Stage-Gate resulted in significant improvements in performance while only improving existing Stage-Gate practices resulted to only little or no improvements. Ultimately, Cooper (2016) highlights that adopting Scrum within this framework does not require abandoning Stage-Gate but rather the two can be integrated into a coherent hybrid system that gains benefits from both.

HPM enables alignment with dynamic stakeholder needs, client contracts, organizational cultures and processes and industry regulations while simultaneously gaining the benefits of agile practices (Gemino et al., 2021; Székely et al., 2025). Consequently, instead of a compromise, HPM represents deliberate integration of complementary project management methods to address the diverse and often conflicting demands of modern project environments (Reiff & Schlegel, 2022; Székely et al., 2025). However, Bianchi et al.

(2020) points out there is an ongoing debate over whether the two methodologies are compatible and how they can best be combined to achieve the benefits from both. There is still no established theory for defining the optimal combination of Agile and the Stage-Gate model, and current evidence remains mostly anecdotal (Bianchi et al., 2020). Nevertheless, Bianchi et al. (2020) highlight that evidence suggests that hybrid models can outperform purely Agile or traditional methods in certain contexts.

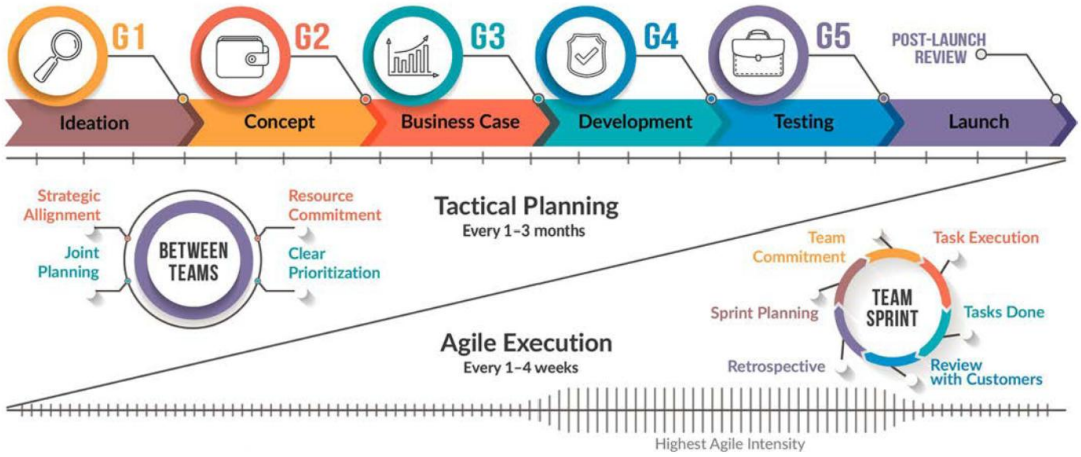

Figure 5. Agile-Stage-Gate model (Cooper & Sommer, 2018)

2.3.1 Strengths and challenges of hybrid project management

HPM models have gained popularity to gain benefits from both TPM methods and Agile practices while mitigating their weaknesses to improve project performance across multiple dimensions. Significant benefits come from the adaptability of hybrid models since it allows organizations to tailor hybrid models and their methods according to specific context and project types (Reiff & Schlegel, 2022; Székely et al., 2025). When choosing the most suitable method and technique for specific project and its phase to meet project and stakeholder needs, risk can be managed effectively, project goals can be achieved faster and with lower costs and overall project outcomes can be improved (Reiff & Schlegel, 2022; Székely et al., 2025). Moreover, one of the key benefits is that HPM models enable senior management to monitor project status and progress from a strategic perspective while allowing development teams to maintain adaptive execution

at the operational level (Zasa et al., 2021). According to Cooper & Sommer (2018) the Stage-Gate model in HPM is more flexible than the traditional Stage-Gate model used on its own. Gates remain essential for strategic decision-making by providing critical go/kill decision points and ensuring alignment with organizational objectives (Cooper & Sommer, 2018). Stages offer high-level overview, structure and deliverables but the deliverables are leaner, less granular and more tangible, such as product design and or prototypes instead of heavy documentation and reports (Cooper & Sommer, 2018). Bianchi et al. (2020) also note that the freezing of the product design is not as strict with Agile-Stage-Gate as it's in Stage-Gate model to enable design modifications for as long as feasible. Agile practices are used for microplanning, daily work control and progress reporting, and essentially enabling iterative development (Cooper, 2023, 2016; Edwards et al., 2019). Reiff & Schlegel (2022) emphasize that with HPM the objectives of the NPD project can be concretized iteratively step by step while maintaining long-term planning, including triple constraints and milestones. This balance facilitates long-term planning, risk management and resource allocation and simultaneously increases efficiency, product development performance and delivery speed (Ciric et al., 2018; Ciric Lalic et al., 2022; Conforto & Amaral, 2016; Cooper & Sommer, 2018; Székely et al., 2025; Zasa et al., 2021). Moreover, HPM improves responsiveness and adaptability to changing market conditions and customer needs, which in turn improve overall project focus and prioritization, ultimately leading to better alignment with customer requirements and higher customer satisfaction (Ciric Lalic et al., 2022; Cooper & Sommer, 2018; Gemino et al., 2021; Reiff & Schlegel, 2022; Székely et al., 2025; Zasa et al., 2021). In addition, HPM facilitates project teams working autonomously and improves communication and coordination among team members, leading to motivated team members with high team morale (Cooper & Sommer, 2018; Reiff & Schlegel, 2022; Zasa et al., 2021).

Reiff and Schlegel (2022) note that many of the benefits of HPM are aligned with the benefits associated with Agile approach. Reiff and Schlegel (2022) write that considering this, HMP appears to offer significant improvements compared to TPM but not compared to APM. Nonetheless, the value of the hybrid approach lies in its ability to

integrate certain agile practices to organizations where adopting a pure Agile approach is constrained (Reiff & Schlegel, 2022). This is the case particularly for many large organizations with large-scale projects working in industries with strict compliance requirements (Sithambaram et al., 2021; West et al., 2011; Zasa et al., 2021). HPM is therefore particularly suitable for larger organizations and complex projects, enabling these organizations to adopt agile practices that may otherwise be impractical and thus gain benefits from them (Reiff & Schlegel, 2022). Some studies indicate that HPM can achieve the same budget, time, scope and quality outcomes as TPM while simultaneously achieving stakeholder success levels similar to those achieved with APM (Gemino et al., 2021; Reiff & Schlegel, 2022). According to Mirzaei et al. (2025), HPM offers greater scalability than Agile methodologies while remaining less costly than TPM. Moreover, according to Reiff and Schlegel (2022), new requirements or changes in prioritizations can be integrated flexibly without needing to do a complete rescheduling of the project. Concurrently, HPM has been found to mitigate risks in projects with high uncertainty compared to pure an Agile approach (Reiff & Schlegel, 2022). Gemino et al. (2021) write that ultimately HPM has been found to be comparable in effectiveness to a fully Agile methodology. Collectively, these findings suggest that HPM can achieve overall better outcomes and results, especially when compared to TPM methodologies (Reiff & Schlegel, 2022; Székely et al., 2025)

Table 5. Strengths of HPM

Strengths	**Sources**
Business: • Enables Agile adoption in organization with compliance constraints • Suitable for large organizations and complex projects • Improves alignment with customer requirements and satisfaction • Greater scalability than Agile, lower costs than TPM	Ciric Lalic et al. (2022); Cooper & Sommer (2018); Gemino et al. (2021); Reiff & Schlegel (2022); Sithambaram et al. (2021); Székely et al. (2025); West et al.

• Comparable effectiveness to Agile and better outcomes than TPM • Integration of new requirements without full rescheduling • Mitigates risks in uncertain projects	(2011); Zasa et al. (2021)
Managerial: • Combines strategic oversight with operational agility • Flexible Stage-Gate model with leaner, more tangible deliverables • Provides critical go/kill decision points for strategic control • Improves long-term planning, resource allocation and risk management •Improves project focus and prioritization	Bianchi et al. (2020); Ciric et al. (2018); Ciric Lalic et al. (2022); Conforto & Amaral (2016); Cooper & Sommer (2018); Gemino et al. (2021); Reiff & Schlegel (2022); Székely et al. (2025); Zasa et al. (2021)
Product and process: • High adaptability to specific organizational context and project type • Facilitates iterative product design and prototyping • Balances long-term planning with step-by-step iterative execution • Improves responsiveness to changing market and customer needs • Improves product development performance and delivery speed	Bianchi et al. (2020); Ciric et al. (2018); Ciric Lalic et al. (2022); Conforto & Amaral (2016); Cooper & Sommer (2018); Gemino et al. (2021); Reiff & Schlegel (2022); Székely et al. (2025); Zasa et al. (2021)
Team: • Improves communication and coordination within teams • Improves team autonomy • Increases motivation and morale	Cooper & Sommer (2018); Reiff & Schlegel (2022); Zasa et al. (2021)

• Encourages collaboration between management and development teams	

While these benefits highlight the potential of HPM, there are still significant challenges affecting its implementation and adoption in organizations. One of the main challenges comes from the fact that APM and TMP methodologies offer opposite and in many ways contradictory guidelines and recommendations in aspects such as planning, team structure, management style and coordination (Mirzaei et al., 2025). Whereas agile emphasizes flexibility, iterative planning and empowered autonomous teams, plan-driven models are based on predictability, control and hierarchical decision-making. Additionally, Silvius-Zuchi and Silvius (2024) write that since the two approaches have different roles and team structures, creating role clarity and integrating the roles is challenging in HPM. Silvius-Zuchi and Silvius (2024) highlight that role clarity is essential because it directly affects team performance, but there is no one correct solution to achieve the clarity. Overall, the coexistence and the contradictions between the two approaches can cause tensions at all organizational levels (Bianchi et al., 2020).

The implementation of Agile-Stage-Gate model in particular has proven complex. Zasa et al. (2021) note that the ideal structure of Agile-Stage-Gate model remains unclear and that effectively integrating Stage-Gate model with agile practices in practice is complex, even when application levels are separated. Companies often struggle with divergent planning cycles, difficulty in allocating resources to dedicated agile teams and management scepticism (Cooper & Sommer, 2018; Zasa et al., 2021). Additional challenges arise from the characteristics of physical product development, such as producing modular outputs with short sprints, fluid versus fixed product definitions and the need for long-term planning in areas such as budgeting and supplier contracts (Cooper & Sommer, 2018; Zasa et al., 2021). Zasa et al. (2021) also emphasize possible external constraints when external stakeholders require fixed requirements and budgets which makes collaboration challenging when agile practices introduce uncertainty and evolving goals. Beyond these structural challenges, cultural barriers also significantly affect the

implementation of HPM. According to Zasa et al. (2021), shifting organizational mindsets to accommodate the new flexibility that agile practices bring often proves to be difficult. Reiff & Schlegel (2022) also note that the adoption of HPM requires a high degree of methodological competence among all individuals, extensive training, strong communication across teams and different interfaces to achieve high transparency, all of which can create additional administrative effort.

The diversity of different HPM models further complicates their implementation. Reiff and Schlegel (2022) and Székely et al. (2025) point out that differences and similarities among HPM models are still poorly understood and there is still limited knowledge about the prerequisites and success factors for their effective adoption. Consequently, organizations often face challenges when attempting to customize a suitable HPM model for their context and selecting the right tools to use (Reiff & Schlegel, 2022). Moreover, aliging hybrid practices with existing corporate reward systems, keeping agile teams connected to the rest of the organization and overcoming excessive bureauracy remain practical challenges in implemenating Agile-Stage-Gate model.

It's evident that HPM offers both opportunities to gain signifcant benefits and challenges when combining the opposing principles of Agile and TPM approaches, primaly Stage-Gate model. Because of the fundamentally different approaches for managing NPD projects, their combined use may create tensions with negative effect on performance (Bianchi et al., 2020). Furthermore, Bianchi et al. (2020) write that if not considered carefully and systematically, the combination of Stage-Gate and Agile can cause inconsistencies. At the same time, the complementarities of the two approaches are also highlighed as Stage-Gate and agile practices are used at different levels which allows to gain benefits from both simultaneously (Bianchi et al., 2020 ; Edwards et al., 2019). Because of the complexity of HPM approaches and the need for contextual customization, Mirzaei et al. (2025) emphasize the criticality of project manager's role for succesfull adoption and implementation.

Table 6. Challenges of HPM

Challenges	Sources
Business: • Contradictory principles between Agile and TPM create organizational tension. • Cultural barriers and difficulty in shifting organizational mindset toward Agile flexibility • Lack of clear prerequisites and success factors for HPM adoption complicates implementation • Complexity of aligning hybrid practices with corporate structures, reward systems, and existing processes	Bianchi et al., 2020 ; Mirzaei et al., 2025; Reiff & Schlegel (2022); Zasa et al. (2021)
Managerial: • Lack of clarity on ideal Agile–Stage-Gate structure and integration methods • Management scepticism and resistance to new models • High methodological competence, training, and strong communication required for successful adoption • difficulty maintaining agile team connection to the wider organization • Criticality of skilled project management for successful implementation	Zasa et al., 2021; Cooper & Sommer, 2018; Mirzaei et al. (2025); Reiff & Schlegel (2022); Zasa et al. (2021
Product and process: • Challenges in aligning divergent planning cycles, allocating resources, and handling modular outputs. • External constraints such as fixed budgets and requirements limit flexibility.	Bianchi et al. (2020); Cooper & Sommer, 2018; Zasa et al., 2021

• Challenges in producing modular outputs with short sprints • Potential inconsistencies when Stage-Gate and Agile are not systematically aligned	
Team: • Lack of role clarity due to differing team structures in Agile and TPM approaches • Cultural barriers and the need for extensive methodological competence and communication.	Reiff & Schlegel, 2022; Silvius-Zuchi & Silvius, 2024 ; Zasa et al., 2021

2.4 Project success

The understanding of project success has evolved to recognise that projects differ in complexity, size, value, length and type, resulting in differing opinions regarding project success criteria (Ciric Lalic et al., 2022). According to Ciric Lalic et al. (2022), success factors have progressed from focusing primarily on project efficiency and on time, scope and cost to contemporary approaches that view project success as multidimensional, comprising organizational impact, long-term effectiveness and stakeholder satisfaction.

Although best practices and different project success indicators have been identified, according to Rezvani and Khosravi (2019), the unique characteristics of complex projects have not been considered, creating an unclear view of success and failure factors in complex projects. Most complex large-scale projects fail to meet the set objectives, resulting in significant schedule and cost overruns (Rezvani & Khosravi, 2019). In addition to large scope, timeline and budget, complex projects often involve multiple stakeholders with differing objectives, advanced technologies, significant dependencies and a high level of uncertainty (Rezvani & Khosravi, 2019). These characteristics make complex projects significantly more difficult to manage compared to smaller projects. Rezvani and Khosravi (2019) found that the most essential success factors of complex large-scale projects are stakeholder satisfaction, open communication, clear project plan, meeting customer requirements and top management's support. Additionally, some other success factors

identified are training, meeting design goals, problem-solving and staff commitment (Rezvani & Khosravi, 2019). In addition to success factors, Rezvani and Khosravi (2019) emphasizes the importance of understanding and identifying potential failure factors in complex projects to be able to perform in uncertain and changing environments. The three most critical failure factors identified are poor communication, poor project planning and unclear or changing project requirements (Rezvani & Khosravi, 2019). Some other failure factors identified are ineffective leadership, instability of project team and poor project control and risk management (Rezvani & Khosravi, 2019).

According to Mirzaei et al. (2025), project managers' perception of success influences the choice of PMM they use for a project. Some project managers focus on achieving project management success which emphasizes completing project within the predefined time, budget and scope (Mirzaei et al., 2025). On the other hand, others prioritize project success which aims to achieve outcome-oriented goals such as maximizing customer satisfaction and long-term value (Mirzaei et al., 2025). Aligning project complexity and goals, strategic objectives and organizational structure with the chosen methodology is critical since ineffective project management has been found as a major cause of failure, particularly in NPD projects (Ciric Lalic et al., 2022; Mirzaei et al., 2025). However, Ciric Lalic et al. (2022) note that choosing the most suitable PMM to achieve project success can be complex since there are many different, and sometimes overlapping, methodologies.

3 Methodology

This study was conducted for a case company to gain understanding of the current state of NPD project planning, execution and management, compare it with the theory from literature review and find development areas where actions can be proposed. This chapter explains the structure and execution of the study that was conducted with the aim of answering the research questions. A mixed research method was used by conducting semi-structured interviews with both open-ended questions and numerical 1-5 questions, with the emphasis being on qualitative research methods.

3.1 Mixed methods research

Mixed methods research combines both qualitative and quantitative research approaches to provide a more comprehensive analysis and understanding of the research problem (Grafton et al., 2011; Hammond & Wellington, 2021; Saunders et al., 2023). It enables researchers to utilize more than one method to avoid their individual limitations, which is particularly valuable when addressing complex research problems that cannot be sufficiently studied with a single method (Matovic & Ovesni, 2021; Mitchell, 2018). A defining characteristic of mixed methods research is triangulation where findings from different data sources and methods confirm, complement and contrast each other to strengthen validity and reliability (Grafton et al., 2011; Hammond & Wellington, 2021; Mitchell, 2018). Hammond and Wellington (2021) emphasize that effective use of this approach requires careful design decisions about which methods to combine, how much weight each is given, and whether they are applied sequentially or concurrently because these choices directly shape data analysis. Overall, the interaction between qualitative and quantitative methods ensures a more delicate analysis and nuanced insights of research problems (Matovic & Ovesni, 2021).

3.1.1 Qualitative research

Qualitative research is described as an approach that aims to explore and understand phenomena within their natural contexts, emphasizing the interpretations and meanings individuals bring to them and assuming multiple truths exist (Marshall et al., 2021). It utilizes non-numerical data, collected with open-ended data collection methods such as interviews, surveys, observation or case studies (Saunders et al., 2023). Data collection in qualitative research can be structured, semi-structured or unstructured (Saunders et al., 2023). Semi-structured interviews are used when researchers want answers to pre-determined topics and questions while allowing interviewees to provide open-ended and elaborated answers (Saunders et al., 2023). Mainly qualitative research method was chosen because of its suitability for this study, gaining knowledge and broad understanding in the topic of NPD management current state, in which people's knowledge and perspective is crucial. Semi-structured interviews were conducted rather than questionnaires because it allows to direct the interview and ask follow-up questions to clarify answers.

3.1.2 Quantitative research

Quantitative research methods are defined by their use of numerical data and statistical analysis to study and understand phenomena (Basias & Pollalis, 2018; Saunders et al., 2023). They are particularly suited for studies involving large datasets where the objective is to test hypotheses, validate theories or identify patters in a systematic manner (Basias & Pollalis, 2018; Saunders et al., 2023). Quantitative research methods can utilize close-ended questionnaires that allow researchers to translate concepts into measurable variables, thereby ensuring systematic compatibility (Basias & Pollalis, 2018; Saris & Gallhofer, 2014; Saunders et al., 2023). Furthermore, Saunders et al. (2023) write that sampling strategies, such as probability sampling, are often applied to promote generalizability, reinforcing the validity of conclusions drawn from numerical analysis. It was chosen to include five numerical 1-5 questions to the interview questions to also enable

directly comparable answers on interviewees views on current state and to examine if there are any significant differences on perspectives between R&D and SC departments.

3.1.3 Data collection

To gain deeper understanding of the current state of NPD project execution and project management in the case company, mixed methods research was chosen and semi-structured interviews were conducted. The emphasis is strongly on qualitative research methods, but in addition quantitative research methods were used to bring valuable insights in topics where qualitative research methods would have not been suitable. Interviewees were chosen based on their title and role in the case company. The aim was to select interviewees from multiple different NPD projects and from various backgrounds and level of experience. It was chosen to limit the interviewees to roles that are in the core of planning, decision-making and management of NPD projects, including the roles of a director, project manager, scrum master, product owner and technical product owner. With the recommendation from the supervisor, also two people leaders from the R&D department were included. Overall, 18 interviewees were contacted by email or Microsoft Teams chat, from which all were interested in participating in an interview. From these 17 led to an interview, with 3 interviews from each of the roles apart from people leaders with 2 interviews conducted. One interviewee didn't answer the quantitative questions as they felt they didn't have enough information to give answers to them, meaning that sample size for quantitative research is 16. Table 7 contains detailed information from the interviews. For anonymity, names have been replaced with numbers.

Table 7. Interview data

ID	Department	Title	Duration
1	Supply chain	Director	29 min
2	R&D	Director	52 min
3	R&D	Director	36 min
4	R&D	People leader	42 min

5	R&D	People leader	36 min
6	R&D	Scrum master	29 min
7	R&D	Scrum master	30 min
8	R&D	Scrum master	29 min
9	R&D	Product owner	43 min
10	R&D	Product owner	47 min
11	R&D	Product owner	58 min
12	R&D	Technical product owner	1 h
13	R&D	Technical product owner	49 min
14	R&D	Technical product owner	51 min
15	Supply chain	Project manager	49 min
16	Supply chain	Project manager	1 h 42 min
17	Supply chain	Project manager	35 min

All interviews were held in Microsoft Teams during June, July and August 2025. Interview questions were shared with the interviewees before the interview to give the opportunity to familiarize with the questions beforehand. One hour meeting was reserved for each meeting. All interviews were recorded and automatic transcription tool was used for the purpose of doing a thorough notetaking and review after the interview. Interview questions were designed to gain broad understanding on how NPD projects are planned and managed in the case company, what methods and tools are used and what strengths and challenges are recognized in the current state. Each question was considered through the research questions to understand the core of NPD project planning and management in the context of the case company and furthermore how it compares to the theory in the literature review. This was mainly done with open-ended questions. Additionally, numerical 1-5 question were included to collect directly comparable data

on the interviewees' views on the current state, mainly how typical different scenarios are. Original interview questions in Finnish can be found in Appendix 1. Depending on the interviewee, some had not familiarized with the interview questions beforehand, some had a quick look through and a few had thoroughly read the questions and thought of the answers. One interviewee answered the interview questions in an email and a short interview was conducted to ask clarifying questions.

3.2 Data analysis

As mixed methods research was conducted, both qualitative and quantitative analysis are done. Thematic analysis method is utilised for text format data from open-ended interview questions and statistical analysis methods are utilised for numerical data from numerical 1-5 questions. This section provides an explanation of the utilized data analysis processes.

3.2.1 Thematic analysis

Thematic analysis is a general analytic method for qualitative data that is designed for identifying and analysing themes and patterns of meaning across a dataset (Braun & Clarke, 2021: Cassell et al., 2018). It offers a systematic and simultaneously flexible approach that involves coding data to identify relevant themes or patterns that can be further analysed for detailed descriptions and explanations (Saunders et al., 2023). In general, thematic analysis can be used for a wide range of research topics and it provides a practical way to manage and understand even large amounts of qualitative data (Cassell et al., 2018; Saunders et al., 2023). It's flexible because it can be used with deductive, inductive or abductive research approaches where analysis is guided by existing theory, delivered directly collected data or both strategies are combined, respectively (Saunders et al., 2023).

According to Saunders et al. (2023), thematic analysis is relatively straightforward yet in practice the process is neither simple nor linear. Instead, the process is concurrent and recursive, involving data analyse already during data collection and reviewing earlier data to refine codes, categories and emerging themes (Saunders et al., 2023). The process of thematic analysis can be divided into five phases:

1. data familiarization
2. data coding
3. initial theme generation
4. theme development and review
5. theme refinement and definition (Braun & Clarke, 2022; Saunders at al. 2023)

The first phase, data familiarization, involves familiarizing with the data by doing transcripts, reading documents and writing notes and summaries (Braun & Clarke, 2021; Saunders et al., 2023). Furthermore, this involves engaging with the data through repeated reading and searching for meanings, themes, and patterns (Braun & Clarke, 2021; Saunders et al., 2023). The second phase, data coding, involves systematically assigning labels, such as words or sentences, to categorize data segments for capturing single concepts or meanings (Braun & Clarke, 2021; Saunders et al., 2023). This organizes the data into manageable segments, from which relevant meanings can be understood and broader themes or patterns can be identified (Saunders et al., 2023). In the third and fourth phases, theme generation, and development and review, codes are examined and grouped into potential themes, which are then evaluated and refined as connection and distinctions across the dataset are considered (Braun & Clarke, 2021; Saunders et al., 2023). The last phase before writing the analysis, theme refinement and definition, focuses on assessing the coherence of each theme, adjusting or reorganising them where necessary and assigning clear and descriptive labels to capture their meaning (Saunders et al., 2023).

Thematic analysis was chosen for analysing the qualitative data from the interviews because of its suitability to managing data, identifying relevant themes and understanding different meanings, connections and broader themes. An inductive approach is taken as themes are identified directly from the data instead of using an existing framework of themes to examine the data.

3.2.2 Quantitative analysis

Quantitative analysis utilises mathematical and statistical analysis techniques, from basic graphs and tables to advanced techniques such as calculating and visualising statistical relationships between variables and conducting complex statistical modelling (Saunders et al., 2023). Calculations, graphs and tables are typically done with analysis software, such as Microsoft Excel, IBM SPSS Statistics, but it is critical to understand different types of data and to be able to choose an appropriate statistical analysis technique to conduct a valuable analysis (Saunders et al., 2023).

Five numerical 1-5 questions were included in the interview questions. As one interviewee decided not to answer to quantitative questions, the sample size for quantitative analysis is 16. For consistency across the data set, questions 6 and 8 were reverse coded so that higher values represent more positive or favourable outcomes. All interview questions can be seen from Appendix 1. For each question, a bar chart was made in Microsoft Excel, showing the frequency of each numerical value and the mode of the distribution was identified simultaneously. Furthermore, a two-sample T-test was conducted in Minitab data analysis software to test if there are any significant differences between the answers from R&D or SC departments. It's important to note that while the sample size for R&D department was 12, for SC department sample size was only 4, which affects reliability of the test because one interviewee's answer can significantly influence the results. Nevertheless, the two-sample T-test was conducted to provide exploratory insight of whether there were notable differences in answers between the R&D and SC departments.

4 Results

This section covers the results from the interviews conducted in a case company. Results are divided into meaningful themes with the aim of understanding current state of NPD project management, its strengths and challenges.

4.1 Project management methodology

Overall, it's important to first note the differentiating views on what project management methodology is used in a case company for NPD projects. Case company executes HPM methodology that combines State-Gate and Scrum. Stage-Gate is used for high level planning and following the product's maturity level with gates, and Scrum is used for task defining and managing daily work. Case company's NPD includes software, electronics and hardware that all execute Scrum framework for managing daily work. R&D department was the first to implement Scrum as a methodology to execute Agile approach, and Stage-Gate model was continued to be used from previous PMM to support long term planning. Although HPM is implemented, only nine interviewees (1, 2, 6, 7, 9, 10, 11, 14, 16) named it and the rest eight interviewees (3, 4, 5, 8, 12, 13, 15, 17) named scrum or LeSS from the Agile methodologies being executed. Despite almost half viewing scrum or LeSS alone as the implemented PMM, later in the interview all interviewees mentioned Stage-Gate model and other planning and management methods related to traditional project management methodologies being utilized. Different product teams might have slightly differentiating ways of utilizing Scrum and ways of working but all teams have product backlog as a base to manage work tasks (4, 5, 17). Interviewee 2 emphasized that they don't see any contradiction in HPM but rather Scrum and Stage-Gate *"go hand in hand"*. Scrum alone would be inadequate and more structure and regulations are needed for manufacturing and data management (2).

10 interviewees (2, 3, 5, 7, 8, 9, 10, 11, 12, 13) noted that the term "project" is not used in Agile and instead NPD is framed and managed though product teams and releases. Interviewees 1, 6, 14, 15, 16 and 17 didn't note the term "project", and particularly

interviewees 1, 14, 15, 16, and 17 have a stronger project mindset. This is explained by Interviewees 1, 15, 16 and 17 being from SC department that has a project manager role and the rest of the interviewees being from R&D department that has the roles of a PO, TPO and scrum master. Additionally, according to interviewee 4, a more traditional project mindset is more likely to be adopted in hardware development due to a stronger use of the Stage-Gate model. In contrast, software development is carried out through continuous releases where the Stage-Gate model has a less significant role (4).

4.2 Project initiation and clarity of goals

On high level, project initiation starts from the case company's strategy, product portfolio and vision. Product roadmap is created based on customer interface and customer needs which are combined with the case company's own strategy and goals. From the roadmap, PO defines product goals and organises them into a priority order and assigns them to the following releases. PO makes detailed release plans in cooperation with TPO and product team. Releases are done two times a year and release plans are done for two releases ahead which are communicated forward to stakeholders and customers. Based on the release plan, sprint plans are done for 3-6 weeks at a time to allocate tasks to certain team members, managing daily work and ensuring the most important things are done. This strategic planning process is pictured in Figure 6, showing how prioritization and planning is done from top down, and progress and status delivery is created bottom up. In addition, there's a so called "gray side" in release plans where initial product goals are defined for further releases, but more detailed planning hasn't been started (11). Interviewee 11 explains that these are not communicated forward to avoid locking plans too far ahead and to keep flexibility to changes.

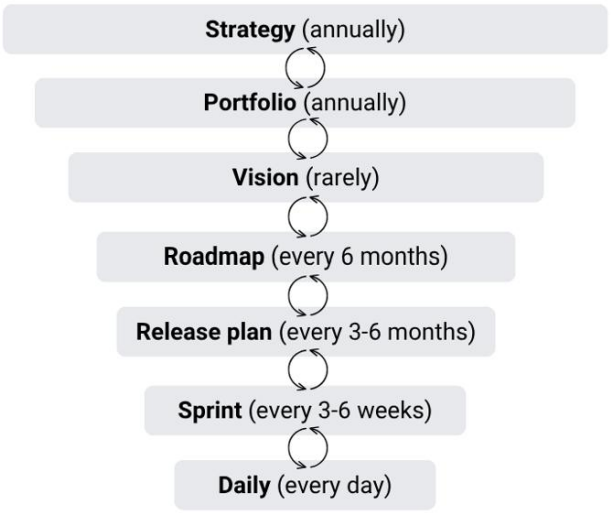

Figure 6. Strategic planning process (adapted from case company process chart)

Officially, instead of projects, releases are executed to develop products iteratively. The Stage-Gate model and its milestone gates are used to support NPD by providing structure and specific deliverables that need to be completed to ensure product maturity level for next milestone. Case company's milestone plan is based on the Stage-Gate model and it includes own milestone deliverables for product ownership, product development, supply chain, and sales, marketing and service. Milestone plan is utilized flexibly, deciding which deliverables are relevant in a particular release. Figure 7 shows the high-level deliverables required from each domain in a particular milestone. M0-M1 planning is the phase in which project/release is defined: fact pack, business case, product concept, technology readiness assessment and preliminary release plan. M0-M1 is a significant stage specially for R&D department. SC department does their own planning work related to supply chain strategy and milestone recommendation during M0-M1, but for SC department M3 is most significant stage when actual development, prototyping and manufacturing of the product is started. Fact pack is formed during M0-M1. Approving the fact pack and freezing product concept from M2 at gate 3 starts M3 which is when actual product development is started. Interviewee 4 points out that milestone plan is emphasized more strongly in hardware development which utilizes a more traditional project mindset while software development is done with a stronger emphasis on continuous iterations and releases. Interviewee 16 highlighted that planning is often started

"under the radar" before officially starting M0 planning. This causes challenges because a small number of people have done much of the groundwork already which restricts the official planning, increasing the risk of the project going in a wrong direction from the start (16).

Case company's milestone deliverables and documentation plan based on the Stage -Gate model

Domain	Delivery type	M0	M1/2	M1	M2	M3	M4	M5	M6
Product Ownership	NPD enablers (develop products)	**-Fact pack** -Business case -Release and integration plan Milestone & documentation plan (+other product related planning tasks)	-Critical customer requirements confirmed	-Risk assessment -Stakeholder & communication plan -Team roles and responsibilities -Updated versions from M0 deliverables (+other product related planning tasks)		-Updated versions from M0-M1 -Project change management -Export control	-Updated milestone deliverables -Updated release & integration plan	**-Plan for transfer to operations** -market quality group review (+Updates to specific previous milestone deliverables)	-Updated business case **-Handover closed**
Product development	Product artifacts (develop products)	-Technology readiness assessment -Concept idea and reuse plan (+other product related planning tasks)		-Technology readiness assessment -Product requirement specification -Product concept (+other product related planning tasks)	**-Product concept freeze** -Updated versions from M1 plan drafts	**-Product design freeze** -Design documentation -DFMEA -Product lifetime estimation -Updated versions from M1 (+other product related tasks)	-Pilot series completed & verified -Status on readiness for Q-release -Updated project quality & documentation plan -Release for field test recommendation	-Product audit plan **-Full production**	**-Product in continuous release** -Updated released product specification -Updated quality & documentation plan
Supply chain	Product manufacturing (deliver products)	-Supply chain strategy -Implementation plan -Investments & budget definition -Sourcing & supplier review -Create SCRS -Milestone recommendation (+other SC related planning tasks)	-Critical production requirements confirmed	-Supply chain concept -SC project budget, CAPEX -Update SCRS task list -Milestone recommendation (+other SC related planning tasks)		-Supply chain design -Supplier evaluation, approval & agreement -PFMEA -Ramp-up plan -Plan for accessories and and spare-parts -Updated CAPEX -Milestone recommendation	-System readiness validated	-Second source qualification -Updated SCRS, CAPEX, PFMEA, ramp-up plan -Milestone recommendation	**-Handover closed** -Market quality report -Milestone recommendation
Sales, marketing & service	Product launch (market and sell products)	-Plan customer involvement -Letter of intent		-Value proposition -Launch goals & strategy -Service strategy (+other product market related planning)		-Global & local launch plans -Plan with all supporting functions -Field test plan	-Status on global & local launch activities	-Updated customer requirement specification, global & local launch plans -Launch tracking -Launch ramp-up readiness	-Market quality report

Figure 7. Milestone deliverables plan (adapted from case company's milestone plan)

Interviewee 11 highlighted challenges in aligning milestones from the Stage-Gate model with iterative release plans, particularly because release plans can be modified along the way. Milestone gates and release dates are not synchronized which causes confusion (11). While milestone gates provide clarity when starting to develop a new product, the following iterative development makes it difficult to define milestones consistently (11). According to interviewee 11, It's unclear which product goals should follow the Stage-Gate model and how milestones should be defined. Furthermore, hardware and software development schedules were seen as difficult to synchronize since hardware development requires longer lead times than software development.

According to the majority of interviewees, project objective is fairly clear (Figure 8) in the beginning of a project (1, 2, 3, 4, 6, 7, 8, 9, 12, 13, 14, 15, 17). Many interviewees described that objectives are well-defined in terms of product goals and functionalities,

while acknowledging that details, scope or schedules often evolve during execution (1, 4, 6, 9, 10, 11, 12, 13, 15, 17). Interviewees noted that while the goals are clear, solutions and execution to achieve these goals may change during development (1, 9, 10, 15) and often release plans change due to reprioritizations and customer-related changes (6, 11, 13). Interviewees 12 and 17 particularly highlighted that although objectives are clear in the beginning, changes may also easily lead to scope creep along the way. According to interviewees 11 and 16, the project objective is only somewhat clear in the beginning. Interviewee 16 highlighted that goals related to the product are generally clearer than how the product is carried out, especially considering project schedule, budget and human resources. Overall, interviewees 7 and 11 pointed out that a foundational aspect of Agile approach is being responsive and flexible in changing environments, which is why release plans and schedules often change.

From the two-sample T-test, it was seen that R&D department rated the clarity of a project objective slightly higher (mean=4.00) than SC department (mean=3.75). However, this difference is not statistically significant (T-value=0.90, P-value=0.420), indicating that both departments had a very similar opinion on the clarity of a project objective in the beginning of a project.

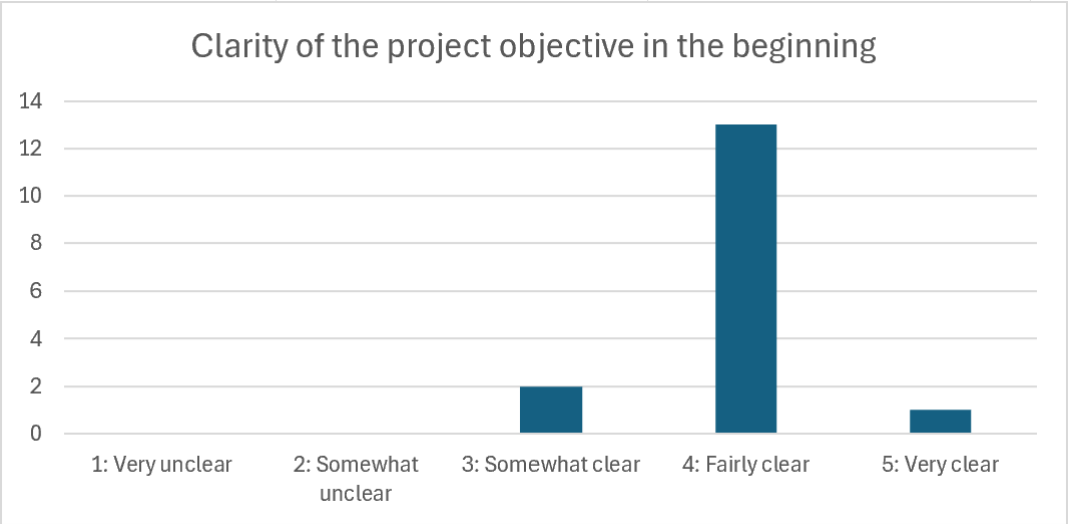

Figure 8. Clarity of the project objective in the beginning

4.3 Scope definition and changes

Project scope is defined based on the market and customer requirements, from which all features, functionalities, product variants and option are defined to product goals and assigned to releases based on their priority. Product owner makes the final decision considering the product and therefore the scope, but SC department is also included from the beginning to give their opinion on the number of variants and how they are possible to manufacture. Interviewee 16 emphasized that scope is closely linked to project initiation and if groundwork is not done carefully and realistically, the scope easily becomes misaligned. Several interviewees noted that scope is left a bit open in the beginning because tasks are not defined in detail (3, 4, 7, 14). However, Stage-Gate model provides general deliverables for NPD that are collected in Milestone document. In addition, R&D has an activity overview list and SC department has a RCPS task list collected from previous projects that contain more detailed task lists for each milestone.

Interviewees' responses about the commonness of scope changing during a project varied considerably (Figure 9). Majority of the interviewees considered scope changes happening very often (7, 10, 11, 12, 13, 14, 16), noting that change is the only certainty in projects (7, 12, 13). Interviewee 16 highlighted that in addition to scope creep being typical, long NPD projects may require rework when new technologies become available on the market or when standards change. Others rated scope changes happening often (1, 4, 6, 9), describing customer requirements changing often (4, 9) and release content changing because of prioritization (4, 6). Some responded scope changing only sometimes (2, 3, 8, 17) or rarely (15), also mainly describing changes coming from changing customer requirements (2, 3). Interviewees 1 and 17 highlighted that previously it was more common for the scope to change but nowadays scope has been learnt to keep clearer, avoiding bigger scope changes such as new product variants being added during a project (17). In addition, interviewee 16 noted that adding new variants or options in the middle of a project is a bit grey are because then the possibilities to influence the initial planning is limited since certain groundwork has already been done and scope creep comes to the product team as a surprise.

On average from the two-sample T-test, R&D department rated scope changes being more common (mean=1.75) than SC department (mean=2.50). However, the answers from SC department varied considerably, ranging from 1 to 4, which makes the mean less reliable due to the small sample size. Overall, this difference is not statistically significant (T-value=-1.08, P-value=0.358), indicating that department didn't considerable affect the opinion about the commonness of scope changes during projects.

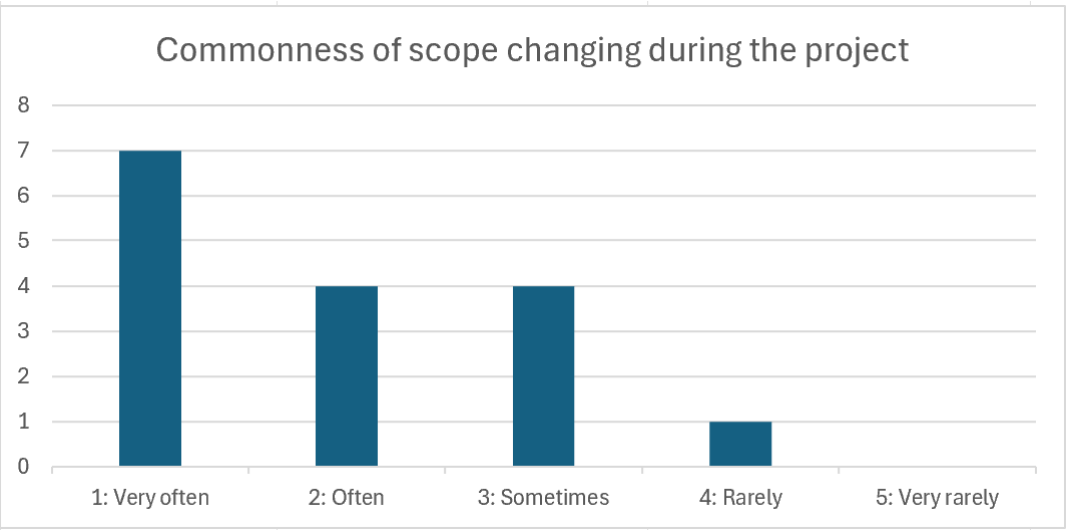

Figure 9. Commonness of scope changing during the project

4.4 Task planning and critical path

Task planning was described as an iterative process, beginning with a high-level project plan with a Gantt-style schedule, which is then progressively refined into detailed sprint plans covering a few sprints at a time (2, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17). Milestones and relevant deliverables are decided on the milestone deliverables and documentation plan for the project. Product backlog is done till the following milestone gate and sprint tasks are further refined from the product backlog for the upcoming sprint. Interviewees 7, 9, and 10 particularly emphasized that by limiting detailed planning to only one or two sprints at a time, unnecessary work in task allocation and scheduling is avoided, as changes are expected to occur regardless. Nevertheless, there is

documentation of general tasks for NPD projects that are followed, which are based on previous projects. SC department has SCRS task list and R&D department has activity overview list, which act as a base for required tasks and when they should be started (1, 4, 5, 6, 8, 9, 11, 15, 17). Figure 10 shows an example of the SCRS list from a particular project's M3 milestone high-level tasks for SC department. Each high-level task has sub-tasks that are scheduled in more detail and allocated for certain team members as sprint tasks. This SCRS list is made in Microsoft Project which also creates a critical path automatically.

Figure 10. M3 from SCRS list (example project from case company)

Several interviewees pointed out that the identification of critical path should be improved (1, 2, 3, 7, 12, 13, 15, 16), some stating that critical path is currently identified very poorly if at all (1, 12). According to interviewee 1, there are still large gaps related to critical path identification, particularly in productization and certification of parts. Furthermore, interviewee 12 highlighted that critical path is largely missing because stakeholders mainly focus on their own work instead of creating shared visibility through collaboration. Additionally, interviewee 12 emphasized that to utilize Polarion for identifying critical path, it is crucial that tasks' workload estimates are precise, and visualization is improved since Polarion is still being developed and learned to use. A step forward to improve the visualization of critical path in Polarion, a Gant chart has been recently added (7, 13). Overall, interviewee 2 noted that when Polarion is taken into use by all stakeholders, tasks dependencies from different stakeholders can be seen, therefore the identification of critical path is improved.

Multiple interviewees explained that critical path is not systematically identified but known in advance from previous experience (6, 8, 9, 13, 14, 17). Known bottlenecks mentioned were testing and validation, human resources and product structure creation (6, 8, 9, 10, 11, 15, 17), and component delivery times particularly in hardware development (14). Interviewee 16 explained that critical path is identified more systematically for new projects, whereas in long-running projects critical path monitoring tends to be less structured and not paid enough focus, depending on the product team. Furthermore, the role of the SC department is often emphasized in critical path monitoring as delays usually become visible during productization when development work is not completed, preventing the SC from starting its work (16).

On the other hand, some interviewees emphasized that identifying the critical path is not relevant in timeboxed releases (4). In addition, while interviewees 11 and 13 noted that identifying the critical path is not part of the Agile approach, interviewee 13 considered this a weakness and interviewee 2 added that in such a large organization dependencies cannot be removed and therefore must be managed. Interviewees 4, 8, 11, 13 and 17 stated that critical path is not identified regularly or systematically and there is no separate review or documentation for it. Critical plan is challenging in Agile approach since it would require more detailed plan from the beginning (7, 11). Moreover, Interviewee 7 pointed out that when tasks change constantly, it's not reasonable to plan too far ahead, therefore critical path is identified continuously in short term. Similarly, interviewee 15 pointed out that although critical path is recognised, it modifies constantly because of changes, confusion and challenges for example in communication. Interviewee 11 highlighted that instead of utilizing critical path for identifying task dependencies, it is typically used on ad hoc basis, for instance when a manager request information on a specific issue or when justification is needed to highlight bottlenecks or resource needs.

The interviewees' perspective on the commonness of project tasks changing during a project varied significantly (Figure 11). The most frequent answer was that project tasks

change often during a project (2, 4, 9, 11, 12, 13, 15, 16), describing that changes to project tasks can come from customer-related reasons, for example if a customer or customer requirements change and therefore reprioritization is done (4, 11). Interviewees 1 and 14 rated that project tasks change very often, with interviewee 7 stating that project tasks are *"certainly and constantly changing".* Interviewee 8 rated that project tasks change sometimes, describing that changes are more common in electrification side than in hardware development. The second most frequent answer was that project tasks change rarely (1, 3, 6, 10, 17). Interviewee 6 noted that R&D process and its activity overview list provide all required tasks and they rarely change and similarly interviewee 17 stated that tasks are well standardized in the SC department.

The two-sample T-test showed that the commonness of project tasks changing during a project was seen slightly more common in R&D department (mean=2.42) than in SC department (mean=3.00). Nevertheless, the difference was not statistically significant (T-value=-0.89, P-value=0.425), indicating that both departments experience similar amount of task changes.

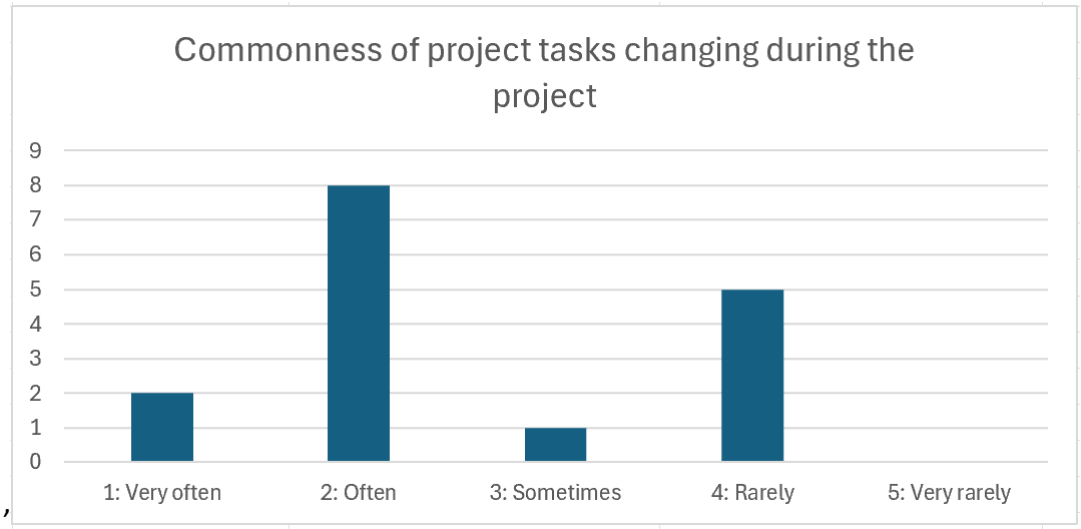

Figure 11. Commonness of project tasks changing during the project

4.5 Project schedule

Project schedules were described being created through a combination of top-down deadlines, planning practices and utilizing different tools. Several interviewees pointed out that deadlines for releases are set by upper management (1, 4, 7, 9, 12, 16), providing only a rough target without detailed knowledge of the required resources or whether the goals are realistic (1, 16). Interviewee 4 highlights that Agile doesn't inherently produce schedule estimates but planning is done according to release dates. Planning is done from top-down, refining product goals further to epics and user stories which are given an estimate of a required workload. Product goals are allocated to releases based on the estimated workload and priority. Lower-priority product goals may be shifted to later releases but if the development of a top-priority product goal is delayed, it's also possible to delay a release date (7). According to interviewee 9, the most common scenario is scope being changed and product goals being moved to later releases, and rarely release date is delayed.

In general, interviewee 8 highlighted that project schedules are often challenging and created assuming that everything will go smoothly which leaves little room for challenges. On the other hand, according to interviewee 3, it is aimed to include a small buffer in case of possible changes. Interviewee 16 explains that since the goals coming from upper management don't usually match with the schedule and resource planning done at the start of a project, the schedule is then tried to get closer to the goals. In practice this means defining what tasks are left out and what shortcuts or risks can be taken intentionally (16).

The responses to how often projects stay on schedule varied significantly from very rarely to often (Figure 12). The most common rating was that project stay on schedule rarely (1, 6, 9, 10, 11, 15, 17). Interviewees 12, 14 and 16 gave a lower rating of project staying on schedule very rarely. Interviewees 2, 3, 4, 7 and 8 had a little more positive perspective, stating that project stay on schedule sometimes and additionally interviewee 13 answered that projects stay on schedule often. Interviewee 17 pointed out

that first plan and schedule that are made are basically only estimates that are specified when more information comes available, therefore is it even possible to make a realistic schedule in the beginning. Additionally, interviewee 3 stated that improvements have been made considering staying on schedule and that they believe more improvements will be made when ways of working are developed and aligned. Identified reasons for project delays are presented in chapter 4.6.

The two-sample T-test showed that R&D department gave a higher rating (mean=2.42) than SC department (mean=1.75), suggesting that R&D department has a more positive perspective on project delays and commonness of projects staying on schedule. However, the result is not statistically significant (T-value=1.85, P-value=0.098), so no firm conclusion can be made.

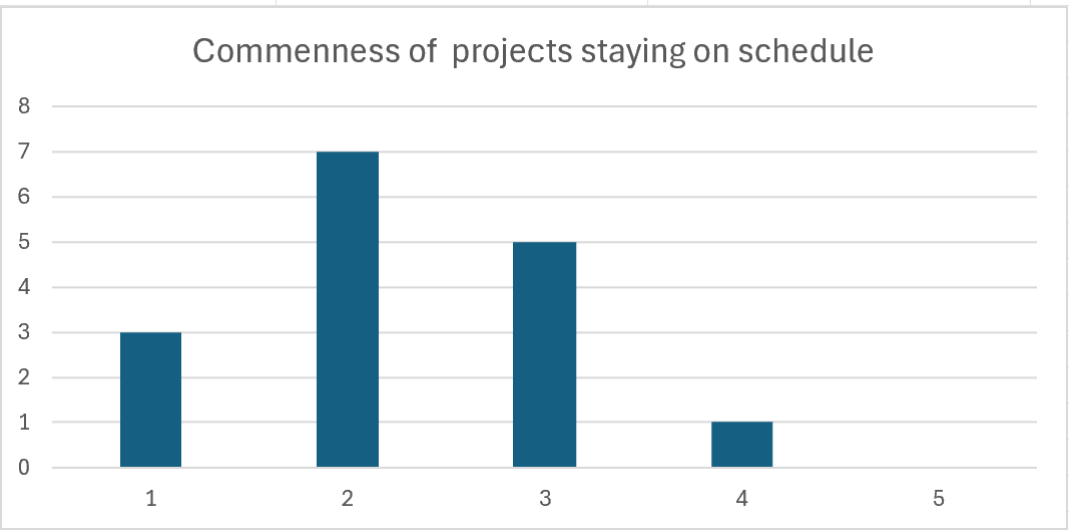

Figure 12. Commonness of projects staying on schedule

Multiple tools are utilized for scheduling but interviewee 9 noted that there are currently no proper tools for schedule-making. Several interviewees mentioned utilizing Microsoft PowerPoint and Miro for high-level planning and visualisation (2, 5, 6, 10, 11, 13, 16, 17). In addition, Microsoft excel is commonly used also for more detailed planning (8, 11, 14, 16, 17). Microsoft Project is also utilized by some to make detailed schedules and listing required tasks and their dependencies (6, 15, 17), but interviewee 17 notes that it still

requires some learning. Software Polarion was frequently mentioned as the main tool for listing tasks, backlog management and release and schedule planning (2, 5, 6, 8, 9, 10, 11, 12, 14, 15, 16) through interviewees 2 and 14 noted that it's not well suited for visualisation. Interviewees 14 and 16 high-lighted that Polarion is used more for listing and assigning tasks and backlog management than for scheduling, interviewee 16 stating that "there is barely any scheduling and its management to even speak of".

The interviewees described multiple ways to identify critical delays in projects with different early warning indicators. Interviewees 13 and 16 stated that a clear early warning indicator is naturally when it is seen straight from the beginning that the schedule is too tight. Interviewee 13 further explained that product team members often bring up that the planned workload is too large, but that they need to be able to justify and communicate why. Many described that continuous product backlog management and monitoring progress with a burn-down chart very quickly shows whether a planned pace is followed or not with completing tasks and achieving deliverables (2, 3, 4, 6, 7, 11, 16). Experience, tacit knowledge and "a feeling" coming from multiple different things indicating that project is not progressing in the right pace are also not-ed by interviewees 11 and 16. Practical things mentioned indicating critical delays were technical challenges that not have been prepared for (14, 15), high number of design changes (15, 16), extra prototype builds (11), delays in critical tests (1, 12), slowness in material creation (1, 16) and external factors such as long approval periods (12). Interviewee 16 highlighted that a long chain of dependencies means that delays in one task inevitably cause further postponements in starting other tasks, for example design changes leading to rework which delays starting of other tasks. Moreover, an-other early warning indicator is attempting to pass a milestone gate despite deficiencies, which results in entering the next milestone already at a disadvantage (16). De-spite above early warning indicators identified, interviewees 9, 14 and 17 stated that critical delays often come as a surprise that could not have been prepared for, for ex-ample customer requirements changing or long component delivery times.

When early warning indicators indicate critical delays in a project, the next question is how they are reacted. Many interviewees described that typically tasks and deliverables are reprioritized, and less critical activities are delayed to adjust scope with available resources and time (3, 6, 7, 10, 11). Moving deadlines was also described as a frequent action when delays cannot be resolved otherwise (1, 7, 11, 13, 17). Simply, one of the project triple constraints must be adjusted (9), and because rarely more resources are available (3, 6, 11, 17) it leaves either scope or time to be adjusted and according to interviewees 11 and 17 it's often scope since certain product goals are delayed to later releases.

Interviewee 17 stated that reaction should be fast but interviewee 16 described that although corrective actions are attempted, in practice the corrective effect is often too small for the detected deficiency, stating that it's "too late, too little". Interviewees 1, 4, 13, 15 and 16 highlighted that often corrective actions are delayed till the last minute. Interviewee 16 described a culture of "needing to hit a wall full speed" before getting proper help and resources to fix things, and at that point the workload for corrective actions is already much larger than it would have been with much earlier actions. Interviewees 15 and 16 pointed out that milestone gates may be attempted to pass despite deficiencies, describing that changes to a product are delayed to be able to pass a gate or a minimum level of maturity is discussed to pass a gate. Although interviewees 4 and 13 also noted that corrective actions are often delayed, they emphasized that a significant reason for this is the high pressure to achieve objectives and deadlines since they come from customer requirements and therefore should not be changed lightly. This causes conflict and difficulty in accepting reality, in a way leading to waiting and wishing for the state of the project to improve (4, 13). Additionally, interviewee 4 noted the challenge to provide the organization with sufficient transparency about the realistic state of a project.

4.6 Project delays

Interviewees identified a wide range of reasons for project delays. All identified reasons for project delays are all collected in the table 8. The most frequently mentioned reason for project delays was lack of resources, which was mentioned all but two interviewees (1, 2, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 17). This includes both human resources as well as testing capacities. Shared resources, particularly test laboratories, were frequently identified as a bottleneck by interviewees 5, 7, 9, 10, 11, 12, 13, 17, which results to project not progressing according to plan (11). Interviewee 13 further emphasized that limited availability of test resources often delays testing which in turn leads to technical problems being discovered in the final stages of the project. Shared human resources, such product structure team and other SC development teams, were specifically mentioned as a bottleneck by interviewees 9 and 11. Directly linked to lack of resources are the challenges coming from prioritization of resources (3, 4, 5, 7, 8, 9, 10, 15). Several interviewees highlighted that when multiple product teams share limited resources, prioritization decisions may lead to delays in certain project as others are given priority (3, 4, 5, 7, 8, 9, 10, 15). In practice, this often means that product teams don't have needed shared resources available at the right time. Interviewee 9 describes that *"it is not understood that prioritization means that something else is deprioritized"*.

Pressure for overly optimistic schedule was seen as a foundational reason for project delays (6, 12, 14 15, 16). The pressure coming from upper management or customers often leads to too tight and optimistic schedule, although it is known already in the beginning that the schedule is not achievable (6, 12, 14, 15, 16). According to interviewee 15, there's a will to develop products quickly but there's not enough resources or ability to prioritize correctly to achieve set schedules. While interviewee 12 noted that tight schedules can make the product team at least try to achieve time objectives by working efficiently, interviewee 16 questioned if there is a real schedule at all if it's unrealistically tight from the beginning. Furthermore, interviewee 16 stated that instead of starting with low resources and increasing them towards the end, if projects would be given more resources at the beginning, it would most likely reduce project lead time, overall

cost and resource needs. In addition to scheduling, delays are also caused by incomplete recognition of all required work (2, 3, 4, 5, 8, 10, 14, 15). Interviewees described challenges in accounting for all stakeholders, all required work and dependencies (3, 4, 5, 8, 14, 15). The importance of putting more effort to groundwork and planning was emphasized (14) as undone tasks that have not been recognized pile up when nearing a milestone gate or end of the project (15). Furthermore, interviewees 2 and 5 highlight lack of communication when required work from stakeholders has been unsuccessfully communicated to them. Another reason for unplanned work is that projects are being executed for the first time with new organization and systems which have brought new procedures, documentation and surprises therefore requiring a lot of learning along the way (2, 9, 10).

Technical problems and technology-related challenges are another recurring reason for project delays (1, 2, 4, 5, 6, 7, 10, 11, 13, 14, 16). Issues such as original technology proving to be unsuitable, unforeseen software complexities or design flaws requiring rework were noted by the interviewees (1, 2, 4, 5, 6, 7, 10, 11, 13, 14, 16). Interviewee 2 also highlighted customer-driven changes to product that can cause rework on design. Furthermore, interviewee 16 pointed out that NPD projects with long lead times involve multiple uncontrollable risks, particularly related to technical challenges. Long lead time of NPD projects, and further delays, may increase the risk of rework on product designs when standards may change during the project or new technology comes to market (16). According to interviewee 16, currently risks are effectively identified but not controlled, and if any major risk is realized the effect is immediately dramatic.

Scope creep and other changes during projects were also mentioned by interviewees 9, 11, 16 and 17. Particularly interviewees 16 and 17 highlighted scope creep as one of the most common and significant reason for project delays and interviewee 16 described that scope creep happening *"very often and very much".* In addition, lack of state of will (1), overall management (5, 8) and transparency in goal tracking (1) were mentioned as

reasons for project delays. Ambiguity related to different ways of working was also pointed out by interviewee 4.

External factors were also identified. Supplier-related challenges, including delays in component availability and quality issues, were noted as reasons to project delays by interviewees 5, 7 and 15. Interviewee 7 also mentioned urgent customer need, such as product update requirements, because they take resources from the ongoing project. Overall, interviewee 8 summarized that the high number of different stakeholders and the complexity of goals in NPD projects challenge staying on schedule. Overall, interviewee 8 summarized that the high number of different stakeholders and the complexity of goals in NPD projects challenge staying on schedule. When multiple of the above issues are realized, the effect of project schedule is multiplying.

Table 8. Reasons for project delays

Reasons for project delays	**Interviewee**
Pressure for overly optimistic schedule	6, 12, 14, 15, 16
Not recognising all related work from all stakeholders	2, 3, 4, 5, 8, 10, 14, 15
Scope creep and changes	9, 11, 16, 17
Lack of resources	1, 2, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 17
Prioritization of resources	3, 4, 5, 7, 8, 9, 10, 15
Lack of overall management	5, 8
Lack of state of will	1
Lack of transparency in goal tracking	1
Complexity of goals	8
Technical problems and technology challenges	1, 2, 4, 5, 6, 7, 10, 11, 13, 14, 16
Lack of communication	2, 5

Ambiguity related to different ways of working	4
Supplier challenges in availability and quality	5, 7, 15
Customer oriented changes during a project	2, 6
Urgent customer needs	7
Multiple uncontrollable NPD related risks	16
Organisational changes and force majeure situations	2, 9, 10

4.7 Multiteam collaboration

Interviewees mostly described that collaboration and communication is mainly managed through Scrum events and with specific weekly or monthly meetings (2, 5, 6, 7, 10, 11, 12, 14, 15, 16, 17). Several interviewees noted the importance of PO forums for aligning goals and resources across product teams (2, 3, 8, 9), although interviewee 9 pointed out that challenges remain since each PO's responsibility is to primarily defend their own product team's priorities. Similarly, interviewee 16 noted that a natural conflict of interest between stakeholders may occur when different stakeholders have differing priorities which may lead to information or requests being overlooked. For instance, if R&D is delayed in their own development tasks, they can be reluctant to allocate time to requests coming from SC department, even when these were necessary for productization (16). Moreover, interviewee 1 emphasized that although collaboration works well within small teams and between individuals, focus on common goals and working together towards them is not on good level.

Interviewee 2 notes that although the structure for collaboration and communication exist within the Scrum events, there is still a lot to improve on how they are utilized. Interviewees 16 and 17 note that transparency between stakeholders is not always on

sufficient level. Related to this interviewee 4 notes that efforts have been made to establish appropriate communication practices and align terminology across departments, but this work is still in its early stages and communication challenges exist due to stakeholders speaking a different language. In addition, interviewee 13 pointed out that sometimes it's difficult to communicate needs to stakeholders on time because it's not always clear or understood what a certain functionality means in terms of the whole product and whether it affects production or not. This is one reason it's important to central stakeholders' representatives, particularly SC department, to participate in meetings and Scrum events actively from the early stages (2, 3, 5, 10, 11, 13, 15, 16, 17). Overall, interviewee 2 points out that although there's still much to improve, it's important to remember that implementing Agile is a big organizational change that takes time to internalize and get all stakeholders onboard.

4.8 Hybrid project management methodology challenges

Overall, the interviewees highlighted several recurring challenges related to Agile and hybrid project management practices in the case company. These challenges are collected in the table 9. The two most frequently mentioned challenges were different ways of working and process gaps (1, 2, 4, 7, 8, 9, 14) and short-term focus and lack of overall management (3, 5, 8, 10, 15, 16, 17). Different ways of working are a directly related to slow Agile and Scrum adoption within the organization (2, 4, 9) since all departments haven't yet adopted Agile and Scrum. Furthermore, according to interviewees 2 and 14, departments adopting Agile and Scrum haven't been able to shake old ways of working which has led to vague hybrid methodology (4, 7, 9, 11, 14) that is executed differently in different product teams (9). This was also descripted as forcing project mindset that is not supported by the executed Agile approach and Scrum framework (4, 7, 9, 11, 14). The forced project mindset comes from the management expecting clear project schedules at the start of a project (4, 14) and therefore the hybrid model is still very project like (9). Interviewees 7 and 11 highlighted that a critical reason why the current hybrid model is difficult to understand and causes confusion is because both Agile product releases and Stage-Gate milestones are used but they are difficult to coordinate and are

not aligned. Furthermore, it's not clear when and when not product goals should go through the Stage-gate milestones once the product is in iterative development phase since the amount of documentation it requires can quickly become a challenge.

Moreover, while product team is responsible for the development of a product, not all responsibilities should be allocated directly to them (10). Interviewee 10 stated that expecting each product team to master every specialized task, such as legal and standard compliance and documentation, leads mediocre results and taking away developers' resources from the actual development work. Overall, having expertise for these kinds of specialised tasks in each product team is inefficient and leads to resource challenges (10). Instead, interviewee 10 notes that more centralized support functions should be utilized to support product teams in tasks that are not central in development. One existing example for this is NPI team that was created for this exact purpose to free product team's resources from documentation and this has led to NPI team making the work more efficient (10).

Short term-focus and lack of overall management was the other most frequently named challenge (3, 5, 8, 10, 15, 16, 17). Interviewees 16 and 17 emphasized the risk that focus shifts primarily to short-term plan and current tasks while the broader overview and long-term objectives become neglected, therefore losing control over overall management. According to interviewee 16, focusing on near future and current activities can result to overlooking dependencies and failing to align short-term work with long-term plans. The siloing of product teams was also mentioned as a significant challenge for achieving overall management (10, 15) and interviewee 10 descripted that siloing results to *"everybody just tinkering with their own things"*. According to interviewee 10, this further results to inefficiency, wasting resources and unnecessary problem-solving when product teams are doing same things without control. Interviewees 8 and 15 also highlighted the importance of everybody using same systems and tools to achieve transparency and interviewee 8 stated that the overall picture is impossible to achieve without common tools and practices.

Additionally, there were some fewer times mentioned challenges that can be associated with short-term focus and lack of overall management. Lack of goal setting and tracking were mentioned as an overall challenge by interviewees 1 and 16. Long-term planning is seen as difficulty (5, 7) and interviewee 5 explained that keeping timelines under control is challenging when details are not clearly defined at the beginning of a project but are instead further defined along the way. This also affects defined scope and schedules changing along the project and interviewee (11) highlighted the challenges this brings to both internal and external stakeholder communication. For example, too early communication to customers might lead to customers disappointing when priorities in release schedules change (11). Understanding the whole project and recognising its related tasks from all stakeholders is limited and deficient (2, 5) which results to unplanned work, particularly when project is nearing a milestone gate. Interviewee 2 highlighted the importance of understanding different stakeholders' work at least in *"common sense level"* to be able to consider all stakeholders' work and at which phase of a project a particular stakeholder's resources are needed.

Unclear roles, responsibilities and project ownership are also seen as a major challenge (1, 6, 9, 14 and 17). Overall, a project manager role with project ownership is missing from Scrum framework (6, 9, 14) as it has been divided between PO, TPO, SM and developers. This is seen as a challenge because there isn't one role or person who has ownership, knowledge and responsibility of the whole project (6, 14). Furthermore, according to interviewees 6 and 14, responsibilities of a PO, TPO and SM differentiate between product teams, and also depend on people's previous experience. This results to some responsibilities *"falling between roles"* (6). Because PO has ownership over product backlog, they are easily thought to have multiple roles simultaneously (9) but according to interviewee 6 PO is somewhat detached from the development team's daily work. In theory it is aimed at that the whole product team together is responsible for the project, but this doesn't work in practice and one role with overall ownership is needed (6, 14).

Some other challenges were mentioned less frequently but nevertheless provide further insights into current challenges. Challenges in cross-team collaboration and communication were highlighted by interviewees 13 and 15. Interviewee 13 explained that this includes the challenge of effectively communicating new ideas and gained knowledge between product teams. Interviewee 15 also highlighted the inefficiency of using overlapping systems which causes double work and doesn't necessarily increase transparency. Related to product teams, a natural conflict of interest was also noted by interviewees 8 and 16 in situations such as prioritization of shared resources. Related to these kind of situations, interviewee 9 states that POs shouldn't be responsible of making shared prioritization decision between product teams, which is the current way of working, as POs should always act in favour of their own product team. Resource constraints and thin product teams were also identified as a challenge because they present challenges into achieving long-term end-to-end teams (2, 13, 14), which are important to efficient product development. According to interviewee 13, there should be at least two specialists of the same topic in each product team to enable discussions and efficient problem-solving. Successful self-management of a product team being very person-dependent was seen as a challenge by interviewees 6, 7, and 16. Sharing responsibility and overall management to team members isn't equally suitable for everyone and some team members prefer more hierarchical way of operating (6, 7). Interviewee 16 describes balancing this as *"a line drawn in water"*. Additionally, difficulties in applying Agile and Scrum practices in hardware development were noted by interviewees 8 and 12 because the work is more long-term than software development and difficult to break down into small enough units for sprints (8). On the other hand, interviewee 12 emphasizes that applying Agile and Scrum practices in hardware development is constantly improving. As an overall challenge, interviewee 11 highlighted constant change since original plans about scope and schedule rarely hold.

Table 9. Challenges of the current PMM

Challenges	Interviewee

Unclear roles, responsibilities and project ownership	1, 6, 9, 14, 17
Short-term focus and lack of overall management	3, 5, 8, 10, 15, 16, 17
Limited understanding of the whole project and its related tasks	2, 5
Lack of goal setting and tracking	1, 16
Different ways of working and process gaps	1, 2, 4, 7, 8, 9, 14
Slow Agile/Scrum adoption and its understanding in the whole organization	2, 4, 9
Vague hybrid model and forced project mindset	4, 7, 9, 11, 14
Difficulty in long-term planning	5, 7
Poor cross-team and stakeholder communication	13, 15
Conflicts of interest between teams	8, 16
Lack of resources and thin product teams	2, 13, 14
Person-dependent self-management	6, 7, 16
Challenges in applying Agile to hardware development	8, 12
Overlapping systems	15
Constant change	11

4.9 Hybrid project management methodology strengths

Several strengths in the current project management methodology were identified by the interviewees. All identified strengths are collected in the table 10. The most frequently mentioned strength was self-directed end-to-end product teams (1, 2, 3, 4, 5, 6, 7, 8, 9). Responsibility can be shared with self-directed product teams to give them

freedom to execute work tasks in a way they see the most fit (1, 4, 5, 6, 7, 8). Furthermore, since end-to-end product teams gain extensive knowledge of the products and its components, they have the best knowledge and understanding of the product and related work tasks (1, 2, 3, 4, 5, 9), therefore can further take future variants into account in the development work (9) and avoid unnecessary work (4). Related to product teams, interviewees 4, 9, 15, 16 and 17 highlighted strong collaboration and communication, which is facilitated particularly by the sprint events of Scrum framework (16). According to interviewee 4, in addition to strong collaboration within product teams, collaboration across team boundaries has also improved significantly. Furthermore, transparency was seen as a significant strength (2, 4, 12, 16) because challenges and problems emerge quickly (2). Interviewee 4 explained that transparency is largely promoted by release plans, product goals and sprint plans. Interviewees 12 and 16 further emphasized using common systems and tools to achieve full transparency across stakeholders by synchronizing all tasks. This is still in progress but improving constantly when more stakeholders take same systems into use (12).

The second most frequently mentioned strength was agility and flexibility (7, 8, 10, 11, 13, 16, 17). HPM model allows reacting quickly to changing conditions and customer requirements which increases customer focus (7, 8, 9, 10, 11, 13, 16) and additionally supports sales ramp-up (11). According to interviewee 13, flexibility helps to avoid unnecessary work because all ongoing tasks are relevant and important to the current priorities. Mitigating unnecessary work also improves efficiency which several interviewees also associated the current methodology (4, 5, 10, 13, 16). The current PMM was described being relevantly lean and simple (16) because certain type of administration and documentation, particularly related to resources and schedules, is omitted (10). Interviewee 5 also emphasized that not everything needs to be known in advance but instead the project is refined progressively when more information becomes available. In addition, interviewee 3, emphasized that PO alone having the ownership over product backlog provides clarity and efficiency in decision-making and prioritization since PO has both the responsibility and freedom to make changes and corrective actions.

Furthermore, there were some less frequently mentioned strengths. In addition to agility and flexibility, interviewee noted that a well-adopted HPM model simultaneously provides clarity and transparency in long-term plan. Continuous releases two times a year has increased value creation to customers, which is seen as significant strengths by interviewees 2 and 11. Furthermore, a valuable aspect of the current PMM is continuous improvement and learning (5, 7, 16), both as a NPD project organization (7, 16) and as individuals having versatile opportunities to challenge themselves (5).

Table 10. Strengths of the current PMM

Strengths	Interviewee
Transparency	2, 4, 12, 16
Agility & flexibility	7, 8, 10, 11, 13, 16, 17
Efficiency	4, 5, 10, 13, 16
Collaboration and communication	4, 9, 15, 16, 17
Continuous improvement and learning	5, 7, 16
Self-directed end-to-end product teams	1, 2, 3, 4, 5, 6, 7, 8, 9
Continuous value creation for customers	2, 11
Clarity of the long-term plan	16
PO owns product backlog	3

When asked to evaluate the overall success of the current PMM on a scale from 1 to 5, the responses varied between 2 and 4. The frequency of responses can be seen from figure 13. The most frequently given rating was that the current PMM is fairly successful (3, 4, 6, 7, 8, 10, 16, 17). Many of these interviewees emphasized that despite its challenges the current PMM is the most effective and overall the best so far and represent a step forward compared to previous PMMs (3, 4, 10, 16). Some of these interviewees summarized that the most significant challenges were related less to the methodology itself and more to resource constraints (6) or person-depended factors (17). Furthermore, unclear roles and responsibilities were emphasized as one of the main challenges

(16). Others pointed out that while the current HPM model is seen as efficient, it is still in a development and learning phase and improvements are ongoing to achieve standardized ways of working (4, 10, 16). Contradictorily, interviewee 16 noted that the current direction might be going too much towards everything needing to be standardized, which is also not preferable. Three interviewees gave a moderate rating of 3, noting that while the current methodology is functional it still requires further development and balance in its practices (2, 12, 13, 15). According to interviewee 2, a certain "*common sense practices"* have been forgotten which has led to a shift from one extreme to another, and a balance is still sought. In contrast, fours interviewees evaluated the methodology more critically with a rating of fairly poor (1, 9, 11, 14), pointing out the vagueness of the current HPM model and variating ways of working (9). Similarly to interviewee 16, interviewee 11 also emphasized significant challenges resulting from unclear roles and responsibilities but associated this directly with product teams not having a project manager role who would have overall responsibility of the project. Furthermore, interviewee 11 emphasized that since there are no longer projects but releases, following the Stage-gate model is challenging in iterative development of new features.

The two-sample T-test showed that both, R&D and SC, departments provided identical means (3.25) when rating the successfulness of the current PMM. The T-test confirmed no difference between departments (T-value=0.00, P-value=1.000), showing that opinions of the current PMM's effectiveness were consistent across both groups.

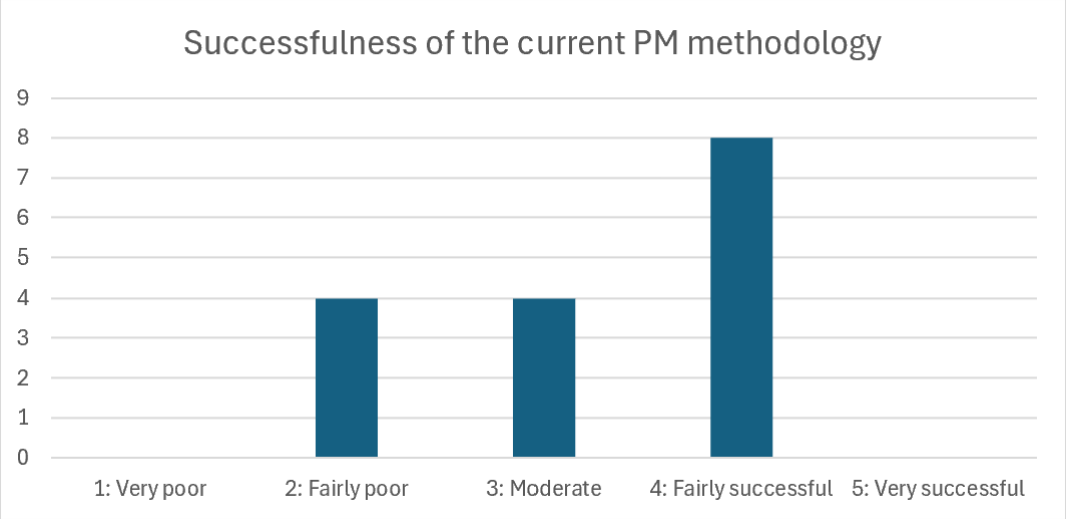

Figure 13. Successfulness of the current PMM

5 Discussion

This section, the results are discussed to make connections between them and further draw conclusions to explore their meaning in the case company. The results are also connected to theory from the literature review. Moreover, managerial recommendations are made to improve the implementation of HPM model and support leadership to identify critical delays in projects with early warning indicators. Overall, this section answers both research questions:

1. What are the challenges of the current hybrid project management model that should be addressed to improve project success?
2. What early warning indicators create transparency to help leadership identify critical delays in projects?

5.1 Key findings

Overall, the current HPM model is seen as the most suitable and effective PMM compared to previous methodologies. However, it is important to note that while many aspects of the current HPM model are executing well, several challenges and areas for improvement remain in its implementation. The hybrid model has allowed the case company to be more adaptive to customer requirements while still providing needed structure and documentation that is required in a big organization operating in a field with strict compliance requirements. According to previous literature, the case company is an exact example of an organization where it's particularly valuable to balance between adaptive innovation and the structure of planning in large and complex NPD projects (Azenha et al., 2021; Cooper & Sommer, 2018; Reiff & Schlegel, 2022; Székely et al., 2025; West et al., 2011; Zasa et al., 2021). Although, large organizations operating in fields with strict governance are most likely to benefits from HPM models, it is pointed out that their practical implementation remains challenging (Cooper & Sommer, 2018; Sithambaram et al., 2021; West et al., 2021; Zasa et al., 2021), which was also pointed out in the interview data. Additionally, an interesting aspect considering the implemented

PMM that should be noted is that around half of the interviewees named Agile, Scrum or LeSS being implemented rather than a hybrid model. This is also recognised in literature, that most Agile implementations are in fact hybrid methods despite many having a perspective of following pure Agile practices (Bianchi et al., 2022; Edwards et al., 2019; Gemino et al., 2021; Reed et al, 2024; West et al., 2011). Furthermore, this indicates a lack of shared understanding of the implemented PMM and execution of the HPM model, which can be best described as Agile-Stage-Gate model based on the different hybrid models described in literature. Overall, the results from the two sample T-tests showing that there is no difference in perspectives between R&D and SC departments indicate that the current challenges are coming from the processes and HPM model itself.

5.1.1 Main challenges of current hybrid project management model

The findings highlight several challenges in the current HPM model that affect the ability to execute projects effectively and successfully. The most significant challenges identified in the case company are short-term focus, lack of overall management, different ways of working, unclear roles and responsibilities and a vague hybrid model together with forced project mindset. Each of these challenges provide insights into how unclear structure and processes of the HPM model can decrease the intended benefits of hybrid approaches. Furthermore, these challenges affect project success in the case company where most common reasons for project delays, according to the interview data, are lack of resources, technology challenges, prioritization of resources and not recognising all related work from all stakeholders.

The identified short-term focus suggests that project activities are managed at an operational level without sufficient focus on long-term plan and coordination towards it. This reflects a disconnect between agile execution on daily activities and centralized project oversight and structure on strategic level. It is noted in literature that organization size, and project size and complexity bring more challenges to applying agile practices, reducing the accuracy of workload estimates and increasing the need for stronger planning structures (Santos & De Cavalho, 2021; Sithambaram et al., 2021; Zasa et al., 2021).

Similarly, the interview data indicated that resource prioritization and late recognition of dependencies often lead to delays, such as when required stakeholder work is not identified early enough. These patterns suggest that there are a lack of common planning and understanding of task dependencies and stakeholders' roles in these tasks.

Another main challenge comes from different ways of working and process gaps. Several interviewees noted that this is a result of slow Agile adoption and different departments implementing different methodologies. Additionally, each stakeholder utilizing different tools restricts overall transparency and information flow of projects, thus complicating maintaining the overview of the big picture. These challenges are also noted in literature that HPM often brings together agile teams and more traditional organizational structures, which can result in inconsistent practices and how work is organized and prioritized (Bianchi et al., 2020; Mirzaei et al., 2025). Additionally, in a large organization with multiple stakeholders, suitable and efficient transparency and information flow requires balancing informal communication with more structured reporting (Boerman et al., 2015). Relying too heavily on face-to-face communication, which is emphasized in APM, may become ineffective and delay sharing critical information across stakeholders (Boerman et al., 2015)

The ambiguity around roles and responsibilities further increases coordination and management issues, highlighting the importance of role clarity for efficient team performance, as emphasized by Silvius-Zuchi and Silvius (2024). Without clear ownership, decision-making becomes scattered and accountability blurred, which can slow down response to emerging issues. The role of a project manager is missed because it provides a clear role that has overall knowledge of the project and responsibility over it. Dividing the responsibilities to PO, TPO and scrum master without clear division of each role's responsibilities has led to project management tasks being handled in a somewhat inconsistent manner with each product team having their own ways of working. This also increases the risk of some responsibilities falling between roles and therefore not being paid enough attention to. Additionally, sharing some of the managerial responsibilities

with team members can pose challenges since some might struggle assuming responsibility and being self-direct (Hoda & Murugesan, 2016), a challenge which was also recognised in the interview data. A potential solution is suggested by the MAHD methodology, which includes a product manager who makes key decisions regarding scope, schedule and resources, while an internal PO ensures the team remains focused on delivering customer value (Simpson & Sheth, 2025).

The vague hybrid model itself and the sense of a forced project mindset indicate a lack of shared understanding of how HPM is executed in practice and what it should achieve. The literature similarly recognises that hybrid adoption often involves cultural misalignments and incompatible processes (Ciric et al., 2019; Hoda & Murugesan, 2016; Atzberger & Paetzold, 2019). Additionally, it's noted that the implementation of Agile-Stage-Gate is complex since it's unclear how to effectively combine Stage-Gate and agile practices even in different application levels (Bianchi et al., 2020; Cooper & Sommer, 2018; Zasa et al., 2021), which is also a challenge in the case company. The interview data also showed the need to find a shared agreement and balance in the use of milestone gates, distinguishing between situations where they provide valuable structure and overall benefits and situations where they only create rigidity and extensive documentation without tangible value. The use of the Stage-Gate model is particularly complex in iterative development, and it is unclear which milestones should be defined and passed again when new iterations are done. This lack of clarity around how agile practices and the Stage-Gate model should be integrated may lead to inconsistent implementation and overall inefficient project management.

Lack of resources, both availability of test laboratories and human resources, and their prioritization were one of the main reasons for project delays. During the interviews, resource challenges were often described particularly being about the availability of resources at the right time. While there might be lack of resources in some aspect or in certain teams, the current unclear implementation of HPM model may also contribute to inefficiencies in how resources are utilized. In particular, siloing of product teams, not

having common understanding of all required work from stakeholders and not recognising task dependencies, combined with short-term focus and lack of overall management can worsen these resource challenges. When shared strategic and long-term planning between product teams and stakeholders is limited, resources need to be prioritized reactively, especially when several product teams would need the same resources simultaneously. Additionally, particularly test laboratory capacity is seen as one of the main challenges affecting timely testing of products, which often leads to technical problems being revealed late in the development process. Similarly, it is noted in literature that hardware testing requires more resources and time than software testing which can pose as a challenge (Atzberger et al., 2019). Based on this, also increasing test laboratory capacity could improve product development process when early testing would reveal technical problems earlier.

Additionally, although product team has ownership over the product and is responsible for all product related activities (Schwaber & Sutherland, 2020), in a large organization it may not be the most efficient use of human resources. This was noted in the interview data, that having all expertise and skills for specific tasks in each product team is inefficient, and rather centralized support functions should be utilized more. Furthermore, this would enable a more flexible use of human resources by allowing them to be flexibly allocated to support different product teams when needed. This perspective aligns with Simpson and Sheth (2025), who note that while scrum teams are cross-functional and dedicated to a single project, MAHD teams operate with both full- and part-time team members across multiple functions, making project resources more flexible. As stated previously, this requires coordinated strategic planning across all product teams, ensuring that resource needs are jointly aligned and managed rather than addressed only within product teams. Overall, it can be concluded that the only root cause for resource challenges may not only be lack of resources but also flexible usage of them and how effectively these resources are planned, coordinated and utilized.

5.1.2 Early warning indicators and reaction to project delays

The findings highlight that early warning indicators have a crucial role in creating transparency and enabling leadership to identify and react to potential project delays. However, their effectiveness depends on how early they surface, how systematically they are monitored and how clearly they are communicated across organizational levels.

One clear early warning indicator identified in the interview data was the recognition of an unrealistically tight schedule already at the initiation and planning stage. This reflects a misalignment between upper management's expectations regarding schedules and resources, and the planning activities carried out by project members. If addressed early, this indicator can provide leadership with a concrete opportunity to realign ambition and capacity before project execution begins. During project execution, continuous backlog management and burn-down chart monitoring are used as structured and quantitative indicators that offer early visibility into velocity of completing planned tasks and staying on schedule. These practices can provide objective evidence of project status and emerging delays to leadership. However, their effective utilization would require common tools between stakeholders to show all work and their dependencies in one place. The interview data also highlighted that providing transparency to upper management about the accurate state of projects remains a challenge. This suggests that the existing challenges in the current HPM model further limit the effective use of early warning indicators and the transparency they could potentially provide for both project teams and leadership. In particular, short-term focus, insufficient recognition of stakeholder work and lack of overall management indicate insufficient oversight of task dependencies, leading to problems being detected too late for timely and effective mitigation.

Attempts to pass milestone gates despite deficiencies further illustrate how early warning indicators might be recognised but not escalated, decreasing their potential to present accurate project status to upper management. It was noted in interview data that high pressure to meet customer requirements discourages early acceptance of risks and postpones decisions. As a result, corrective actions are initiated too late and have limited

impact, often described by interviewees as “too late, too little”. This is particularly problematic when indicators surface late along dependency chains, leaving little time for escalation and corrective actions. This pattern also aligns closely with the previously discussed short-term focus, unclear ownership and lack of overall management, where responsibilities and coordination are scattered. These challenges make it challenging to communicate issues across product teams and organizational layers.

Importantly, these findings are not isolated but directly connected to the structural issues in the HPM model identified in the first part of the discussion. The lack of overall management, different ways of working, and unclear roles directly affect how early warning signals are monitored, recognized and escalated. When ownership is unclear and methods fragmented, early warning indicators remain limited within individual teams rather than being collected and translated into organizational-wide visibility. Therefore, improving project transparency with early warning indicators requires common tools for transparency and also stronger governance structures and clearer accountability, addressing the same challenges that currently limit the effectiveness of hybrid project management.

5.2 Managerial recommendations

Based on the findings and discussion, several managerial actions can be recommended to improve the implementation of the current HPM model and to strengthen leadership’s ability to identify and respond to early warning indicators of project delays. The recommendations focus on addressing the structural issues of current HPM model that currently limit project transparency, coordination and proactive management.

First thing to address would be the lack of common understanding of the current HPM model and the ambiguity of its implementation in all situations. The HPM model should be promoted and trained to all relevant stakeholders to achieve common understanding of how the Stage-Gate model and agile practices support each other and are utilized. A common shared understanding supports following the processes when their purpose

and importance are recognised by all. Foundationally the Stage-Gate model doesn't accept changes during a project, which is something to focus especially when it's combined with agile practices and releases in a HPM model. A clear and structured process model for bringing changes to the Stage-Gate model and analysing the effects within the Stage-Gate model would be essential for improving adaptability and overall utilization of the HPM model. This would be a significant step forward in the effective implementation of HPM model.

A significant reason for project delays in the case company is the late recognition of work required from multiple stakeholders and the dependencies between tasks. To address this issue, all related tasks and their dependencies should be systematically identified as a standard step during project initiation and planning. To achieve this, work has already been started by identifying high level dependencies with a SIPOC chart (supplier, input, process, output, customer). In SIPOC, supplier and customer can mean both internal and external stakeholders. The requirement is that a different stakeholder needs an output from a stakeholder providing it to start their work. Based on the SIPOC chart, a Gant style chart is made to visualize critical path clearly. This may be used as a default critical path for projects but should be examined each time to include possible unique project traits. It should also be noted that the chart only includes high level tasks and thus should be defined at a more detailed level during project planning and at latest when the project is nearing a specific state of execution. Making critical path(s) visible, particularly between different stakeholders, enables more proactive project management, early detection of potential bottlenecks and strengthens leadership's ability to act on early warning indicators before they escalate into critical delays.

Currently, usage of multiple tools and fragmented information flows reduce transparency of project status across teams and organizational levels. A common planning and scheduling platform would provide increased transparency on task dependencies, critical path, workload durn-down chart and overall project status. Polarion as a planned common software represents a step towards shared visibility of overall project status,

but adoption among stakeholders has been slow and it's still in development phase. To fully realize its potential, it is critical to improve Polarion with features that enable effective scheduling and visualization of project progress, as it's currently used mainly for task management rather than comprehensive project management. Overall, increased transparency would not only improve coordination between product teams and stakeholders but also make changes from planned project progress visible to leadership in real time, supporting earlier and more informed reaction.

The results revealed a gap between upper management's expectations regarding schedules and resources, and the project planning work conducted by project managers and product teams. Closing this gap is essential to set realistic targets from the beginning of a project. Allocating sufficient resources at the start of a project can decrease project delays and reduce the need for resource increases later in the project lifecycle. Furthermore, having more realistic schedules and resource plans should be prioritized to ensure more efficient utilization of shared resources between product teams and enable more predictable resource allocation throughout the project lifecycle. Together these actions may not only lower overall project resources, costs and lead time but also free resources to new projects more efficiently.

Strategic resource management is essential to support realistic schedules and prevent cascading delays in projects. This includes coordinated planning of both human resources and test laboratory capacity, which were named common reasons for project delays. Improved scheduling accuracy and transparency of dependencies would allow more effective allocation of shared resources, reducing overlapping demand and bottlenecks. Additionally, increasing the use centralized support functions would make resource utilization more flexible by reducing the dependency of having all required expertise within each product team. Another option could be to utilize part-time team members that can flexibly move between product teams. This would improve responsiveness to changing priorities and help avoid delays caused by having resource constraints and not being able to move resources flexibly where most needed.

5.3 Limitations and future research

The limitations of this study are mainly related to it being a case study conducted in one case company, which limits the sample size and generalization of the results. Although the study provides valuable insights into the practical challenges of implementing HPM model, the results are context specific. Therefore, some of the identified challenges may reflect organizational, cultural or people-related characteristics and issues rather than being directly related to the implemented HPM model. The sample size also presents a limitation. While the number of interviewees was appropriate for the scope of a master's thesis, it further limits the generalization of results. In addition, the sample size for quantitative analysis consisted of 16 interviewees, with an uneven distribution between departments: 4 from SC and 12 from R&D department. This imbalance presents the structural differences between these two departments, since SC department has only the project manager role, whereas R&D department has PO, TPO and scrum master roles. The uneven representation should be noted when interpreting the results, particularly in quantitative analysis.

Another significant limitation relates to the literature used in this study. The benefits and challenges of PMMs are highly context dependent, particularly regarding organizational size, project complexity, and industry uncertainty. While extensive literature exists on Agile methodologies, research specifically addressing their implementation in large manufacturing organizations remains limited. Moreover, studies on HPM models are still largely theoretical, with only few empirical studies conducted. To achieve a comprehensive theoretical framework, literature from other sectors and contexts was therefore included, although all identified benefits and challenges may not be fully applicable to the case company.

The time frame of the study also poses a limitation. Managerial recommendations were developed based on the results and discussion but the effects of implementing these recommendations was not evaluated within the scope of this study. This would require

a follow-up study to examine potential changes in project performance and organizational practices over time.

For future research, it would be valuable to broaden the sample size to include multiple companies. This would allow comparison between different HPM model implementations and provide deeper insights into typical challenges and benefits across contexts. Expanding the sample size would also allow for identifying patterns in how benefits are realized and which practices contribute to successful implementation. Additionally, future studies should consider including team members in the data collection to examine whether their perspectives on the implementation of HPM model differ from those in project management roles. Finally, considering the relevantly broad scope of this study with the aim to identify the main challenges associated with the current implementation of HPM model, future research may benefit from taking a narrower focus. This would enable a more in-depth study of specific aspects of HPM models.

6 Conclusions

This study explored how HPM model can be more effectively adopted to improve project success and how early warning indicators can improve transparency to help leadership identify critical project delays. Together, the literature review and findings from the study provide both theoretical and practical insights into the complexities of implementing HPM in a large organization operating in uncertain environments in a field with strict governance.

The results revealed several challenges that affect successful project execution under the current HPM model. The most significant challenges include short-term focus, lack of overall management, inconsistent ways of working, unclear roles and responsibilities and overall ambiguity of the hybrid model itself. Similar challenges are also recognised in previous literature. These structural issues were also closely linked to the most common reasons for project delays: resource challenges, prioritizations issues, insufficient task recognition from all stakeholders and dependencies of tasks and their critical path. Together these findings highlight the importance of aligning common planning, clear ownership and standardized processes to fully realize the intended benefits of HPM methodologies.

The study also highlighted the role of early warning indicators being critical for improving transparency to accurate project status and enabling timely reaction by leadership. The misalignment between upper management's expectation regarding project schedule and resources, and the project planning work conducted by project managers and product teams is an essential indicator for project delays that should be reacted to. In addition, continuous monitoring of backlogs and burn-down charts of completed work support early detection of risks. However, their effectiveness is constrained by the same issues that limit successful project execution, most importantly short-term focus, insufficient recognition of all project tasks and their dependencies, fragmented communication and transparency across stakeholders.

Based on the findings, managerial recommendations were developed. The findings suggest that strongly promoting the implemented HPM model and sharing common understanding of the roles of both the Stage-Gate model and agile practices is critical for effective implementation. Moreover, change management particularly within the Stage-Gate model is an essential aspect to focus on for its smooth utilization together with Agile releases. Improving the identification of project tasks from all stakeholders and the dependencies of these tasks to visualize critical path would strengthen common planning of resources, proactive management and enable earlier recognition of delays. Improving transparency through common planning and scheduling tools would further support timely detection of delays. Addressing the existing misalignment between upper management's expectations and operational planning is also essential to set realistic schedules and resource allocations from the beginning of a project. Together, these measures would allow leadership to address resource constraints more effectively, improve schedule reliability, and reduce cascading delays across product teams. Additionally, more flexible use of resources through moving part-time product team members or central support functions could increase efficiency and allow developers to focus on their core responsibilities.

Overall, this study adds to the understanding of practical implementation of HPM by examining its benefits and challenges both in theory and in the case company. This study further shows how successful adoption of a PMM is strongly influenced by the organizational context in which it's applied. For the case company, addressing the identified structural issues is essential to taking full advantage of a HPM model, combining the advantages of Agile's flexibility and TPM's structure and governance.

References

Almeida, F., & Espinheira, E. (2022). Adoption of Large-Scale Scrum Practices through the Use of Management 3.0. Informatics, 9(1), 20. https://doi.org/10.3390/informatics9010020

Alqudah, M. & Razali, R. (2016). A Review of Scaling Agile Methods in Large Software Development. *International Journal on Advanced Science Engineering and Information Technology, 6(6), 828. http://dx.doi.org/10.18517/ijaseit.6.6.1374*

Atzberger, A., Gerling, C., Schrof, J., Schmidt, T. S., Weiss, S., & Paetzold, K. (2019). Evolution of the Hype around Agile Hardware Development. *2019 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, 1–8. https://doi.org/10.1109/ICE.2019.8792637

Atzberger, A., & Paetzold, K. (2019). Current Challenges of Agile Hardware Development: What are Still the Pain Points Nowadays? *Proceedings of the Design Society: International Conference on Engineering Design*, *1*(1), 2209–2218. https://doi.org/10.1017/dsi.2019.227

Basias, N., & Pollalis, Y. (2018). Quantitative and qualitative research in business & technology: Justifying a suitable research methodology. *Review of Integrative Business and Economics Research*, 7, 91-105. https://sibresearch.org/uploads/3/4/0/9/34097180/riber_7-s1_sp_h17-083_91-105.pdf

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Martin, R., Mellor, S., Thomas, D., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Schwaber, K. & Sutherland, J. (2001). Agile Manifesto. *Agile Manifesto.* https://www.agilealliance.org/wp-content/uploads/2019/09/agile-manifesto-download-2019.pdf

Almeida, F., & Espinheira, E. (2022). Adoption of Large-Scale Scrum Practices through the Use of Management 3.0. *Informatics*, *9*(1), 20. https://doi.org/10.3390/informatics9010020

Alqudah, M., & Razali, R. (2016). *A-Review-of-Scaling-Agile-Methods-in-Large-Software-Development.pdf*. International Journal on Advanced Science Engineering and Information Technology.

Atzberger, A., Gerling, C., Schrof, J., Schmidt, T. S., Weiss, S., & Paetzold, K. (2019). Evolution of the Hype around Agile Hardware Development. *2019 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, 1–8. https://doi.org/10.1109/ICE.2019.8792637

Atzberger, A., & Paetzold, K. (2019). Current Challenges of Agile Hardware Development: What are Still the Pain Points Nowadays? *Proceedings of the Design Society: International Conference on Engineering Design*, *1*(1), 2209–2218. https://doi.org/10.1017/dsi.2019.227

Braun, V & Clarke, V. (2021). *Thematic Analysis.* SAGE Publications, Ltd. (UK). https://bookshelf.vitalsource.com/books/9781526417299

Bianchi, M. J., Conforto, E. C., Rebentisch, E., Amaral, D. C., Rezende, S. O., & De Pádua, R. (2022). Recommendation of Project Management Practices: A Contribution to Hybrid Models. *IEEE Transactions on Engineering Management*, *69*(6), 3558–3571. https://doi.org/10.1109/TEM.2021.3101179

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. https://doi.org/10.1016/j.jbusres.2018.05.003

Cassell, C., Cunliffe, A. L., & Grandy, G. (Eds.). (2018). *The sage handbook of qualitative business and management research methods : Methods and challenges*. SAGE Publications, Limited.

Ciancarini, P., Kruglov, A., Pedrycz, W., Salikhov, D., & Succi, G. (2022). Issues in the adoption of the scaled agile framework. *Proceedings of the 44th International Conference on Software Engineering: Software Engineering in Practice*, 175–184. https://doi.org/10.1145/3510457.3513028

Ciric, D., Lalic, B., Gracanin, D., Palcic, I., & Zivlak, N. (2018). Agile Project Management in New Product Development and Innovation Processes: Challenges and Benefits Beyond Software Domain. *2018 IEEE International Symposium on Innovation and Entrepreneurship (TEMS-ISIE)*, 1–9. https://doi.org/10.1109/TEMS-ISIE.2018.8478461

Ciric, D., Lalic, B., Gracanin, D., Tasic, N., Delic, M., & Medic, N. (2019). Agile vs. Traditional Approach in Project Management: Strategies, Challenges and Reasons to Introduce Agile. *Procedia Manufacturing*, *39*, 1407–1414. https://doi.org/10.1016/j.promfg.2020.01.314

Ciric Lalic, D., Lalic, B., Delić, M., Gracanin, D., & Stefanovic, D. (2022). How project management approach impact project success? From traditional to agile. *International Journal of Managing Projects in Business*, *15*(3), 494–521. https://doi.org/10.1108/IJMPB-04-2021-0108

Conforto, E. C., & Amaral, D. C. (2016). Agile project management and stage-gate model—A hybrid framework for technology-based companies. *Journal of*

Engineering and Technology Management, *40*, 1–14. https://doi.org/10.1016/j.jengtecman.2016.02.003

Cooper, R. G. (n.d.). The Stage-Gate® System: The Idea-to-Launch Process – An Intro & Summary. *2023*.

Cooper, R. G. (2016). Agile–Stage-Gate Hybrids: The Next Stage for Product DevelopmentBlending Agile and Stage-Gate methods can provide flexibility, speed, and improved communication in new-product development. *Research-Technology Management*, *59*(1), 21–29. https://doi.org/10.1080/08956308.2016.1117317

Cooper, R. G., & Sommer, A. F. (2018). Agile–Stage-Gate for Manufacturers: Changing the Way New Products Are DevelopedIntegrating Agile project management methods into a Stage-Gate system offers both opportunities and challenges. *Research-Technology Management*, *61*(2), 17–26. https://doi.org/10.1080/08956308.2018.1421380

Edwards, K., Cooper, R. G., Vedsmand, T., & Nardelli, G. (2019). Evaluating the Agile-Stage-Gate Hybrid Model: Experiences From Three SME Manufacturing Firms. *International Journal of Innovation and Technology Management*, *16*(08), 1950048. https://doi.org/10.1142/S0219877019500482

Gemino, A., Horner Reich, B., & Serrador, P. M. (2021). Agile, Traditional, and Hybrid Approaches to Project Success: Is Hybrid a Poor Second Choice? *Project Management Journal*, *52*(2), 161–175. https://doi.org/10.1177/8756972820973082

Grafton, J., Lillis, A. & Mahama, H. (2011). Mixed methods Research in accounting. *Qualitative Research in Accounting and Management,* 8(1), 5-21. https://doi-org.proxy.uwasa.fi/10.1108/11766091111124676

Gustavsson, T., & Rönnlund, P. (2013). *Agile adoption at Ericsson hardware product development*. https://doi.org/10.13140/2.1.3781.3447

Hammond, M., & Wellington, J. (2020). *Research methods : The key concepts.* Taylor & Francis Group.

Hoda, R., & Murugesan, L. K. (2016). Multi-level agile project management challenges: A self-organizing team perspective. *Journal of Systems and Software*, *117*, 245–257. https://doi.org/10.1016/j.jss.2016.02.049

Karrenbauer, J., Wiesche, M., & Krcmar, H. (n.d.). *Understanding the Benefits of Agile Software Development in Regulated Environments*.

Lawong, D. A., & Akanfe, O. (2025). Overcoming team challenges in project management: The scrum framework. *Organizational Dynamics*, *54*(1), 101073. https://doi.org/10.1016/j.orgdyn.2024.101073

Maier, N. M., & Emmerich, P. (2022). Agile or Traditional Project Organization: A Quantitative Assessment of Decision Criteria among Firms in the DACH Region. *Central European Business Review*, *11*(5), 67–83. https://doi.org/10.18267/j.cebr.308

Marshall, C., Rossman, G. B., & Blanco, G. L. (2021). *Designing Qualitative Research (7th ed.)*. SAGE Publications, Inc. (US). https://bookshelf.vitalsource.com/books/9781071817384

Matović, N., & Ovesni, K. (2021). Interaction of quantitative and qualitative methodology in mixed methods research: integration and/or combination. *International Journal of Social Research Methodology*, 26(1), 51–65. https://doi-org.proxy.uwasa.fi/10.1080/13645579.2021.1964857

Mirzaei, M., Mabin, V. J., & Zwikael, O. (2025). Customising Hybrid project management methodologies. *Production Planning & Control*, *36*(9), 1188–1205. https://doi.org/10.1080/09537287.2024.2349231

Mitchell, A. (2018). A Review of Mixed Methods, Pragmatism and Abduction Techniques: EJBRM. *Electronic Journal of Business Research Methods*, 16(3), 103-116. https://www.proquest.com/scholarly-journals/review-mixed-methods-pragmatism-abduction/docview/2174222924/se-2

Reed, A. H., Angolia, M. G., East Carolina University, Baham, C., Oklahoma State University, Igah, R., & Oklahoma State University. (2024). Usage of Hybrid Project Management Approaches and Influences on Approach Selection. *Communications of the Association for Information Systems*, *54*(1), 402–427. https://doi.org/10.17705/1CAIS.05415

Reiff, J., & Schlegel, D. (2022). Hybrid project management – a systematic literature review. *International Journal of Information Systems and Project Management*, *10*(2), 45–63. https://doi.org/10.12821/ijispm100203

Santos, P. D. O., & De Carvalho, M. M. (2022). Exploring the challenges and benefits for scaling agile project management to large projects: A review. *Requirements Engineering*, *27*(1), 117–134. https://doi.org/10.1007/s00766-021-00363-3

Sarhadi, P., Naeem, W., Fraser, K., & Wilson, D. (2022). On the application of agile project management techniques, V-model and recent software tools in postgraduate theses supervision. *IFAC-PapersOnLine*, 55(17), 109-114. https://doi-org.proxy.uwasa.fi/10.1016/j.ifacol.2022.09.233

Saris, W. E., & Gallhofer, I. N. (2014). *Design, evaluation, and analysis of questionnaires for survey research*. John Wiley & Sons.

Saunders, M.N. K., Lewis, P., & Thornhill, A. (2023). *Research Methods for Business Students* (9th ed.). Pearson International Content. https://bookshelf.vitalsource.com/books/9781292402741

Silvius-Zuchi, D., & Silvius, G. (2024). Developing role clarity in projects with a hybrid approach. Procedia Computer Science, 239, 904-913.

Sithambaram, J., Nasir, M. H. N. B. M., & Ahmad, R. (2021). Issues and challenges impacting the successful management of agile-hybrid projects: A grounded theory approach. *International Journal of Project Management*, *39*(5), 474–495. https://doi.org/10.1016/j.ijproman.2021.03.002

Sommer, A. F., Hedegaard, C., Dukovska-Popovska, I., & Steger-Jensen, K. (2015). Improved Product Development Performance through Agile/Stage-Gate Hybrids: The Next-Generation Stage-Gate Process? *Research-Technology Management*, *58*(1), 34–45. https://doi.org/10.5437/08956308X5801236

Špundak, M. (2014). Mixed Agile/Traditional Project Management Methodology – Reality or Illusion? *Procedia - Social and Behavioral Sciences*, *119*, 939–948. https://doi.org/10.1016/j.sbspro.2014.03.105

Sutherland, J., & Schwaber, K. (2020). *2020-Scrum-Guide-US.pdf#zoom=100.pdf*. https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf#zoom=100

Székely, B., Késmárki-Gally, S. E., & Lakner, Z. (2025). Hybrid project management: Scoping review. *Project Leadership and Society*, *6*, 100182. https://doi.org/10.1016/j.plas.2025.100182

Thesing, T., Feldmann, C., & Burchardt, M. (2021). Agile versus Waterfall Project Management: Decision Model for Selecting the Appropriate Approach to a Project. *Procedia Computer Science*, *181*, 746–756. https://doi.org/10.1016/j.procs.2021.01.227

Uludag, O., Kleehaus, M., Dreymann, N., Kabelin, C., & Matthes, F. (2019). Investigating the Adoption and Application of Large-Scale Scrum at a German Automobile

Manufacturer. *2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE)*, 22–29. https://doi.org/10.1109/ICGSE.2019.00019

Weichbroth, P. (2022). A Case Study on Implementing Agile Techniques and Practices: Rationale, Benefits, Barriers and Business Implications for Hardware Development. *Applied Sciences*, *12*(17), 8457. https://doi.org/10.3390/app12178457

Wells, H. (2012). How Effective Are Project Management Methodologies? An Explorative Evaluation of Their Benefits in Practice. *Project Management Journal*, *43*(6), 43–58. https://doi.org/10.1002/pmj.21302

West, D., Gilpin, M., Grant, T., & Anderson, A. (2011). Water-scrum-fall is the reality of agile for most organizations today. *Forrester Research*, 26(2011), 1-17. http://www.storycology.com/uploads/1/1/4/9/11495720/water-scrum-fall.pdf

Zasa, F. P., Patrucco, A., & Pellizzoni, E. (2021). Managing the Hybrid Organization: How Can Agile and Traditional Project Management Coexist? *Research-Technology Management*, *64*(1), 54–63. https://doi.org/10.1080/08956308.2021.1843331

Zavyalova, E., Sokolov, D., & Lisovskaya, A. (2020). Agile vs traditional project management approaches: Comparing human resource management architectures. *International Journal of Organizational Analysis*, *28*(5), 1095–1112. https://doi.org/10.1108/IJOA-08-2019-1857

Appendix 1. Interview questionnaire

Tausta:

1. Kerro lyhyesti työtehtävästäsi ja osastosta, jolla työskentelet.
2. Mitä projektihallintamenetelmää/-menetelmiä osastollasi käytetään?

Projektin asetus, suunnittelu ja toteutus:

3. Miten projektit asetetaan ja mitkä vaiheet siihen sisältyy?
4. Kuinka selkeä projektin tavoite on asteikolla 1-5, kun projekti aloitetaan ((1=erittäin epäselvä, 2=melko epäselvä, 3=jokseenkin selkeä, 4= melko selkeä, 5=erittäin selkeä)
5. Miten projektin laajuus määritetään ja mitkä vaiheet siihen sisältyy?
6. Kuinka yleistä asteikolla 1-5 on, että projektin laajuus muuttuu projektin edetessä? (1=todella harvoin, 2=harvoin, 3=joskus, 4=usein, 5=erittäin usein)
7. Miten projektin tehtävät suunnitellaan ja millä tasolla eri aikaväleille?
8. Kuinka yleistä asteikolla 1-5 on, että tehtävät muuttuvat projektin edetessä? (1=todella harvoin, 2=harvoin, 3=joskus, 4=usein, 5=erittäin usein)
9. Tunnistetaanko projektien kriittinen polku? Jos tunnistetaan, millä tavalla ja mitkä vaiheet siihen sisältyy?
10. Miten projektin aikataulu luodaan ja mitä työkaluja ja tapoja siihen käytetään?
11. Kuinka usein projektit pysyvät suunnitellussa aikataulussa asteikolla 1-5? (1=todella harvoin, 2=harvoin, 3=joskus, 4=usein, 5=erittäin usein)
12. Mitkä ovat yleisimpiä tunnistettuja syitä projektien aikataulujen myöhästymiseen?
13. Mitä varhaisvaroitusindikaattoreita käytetään tunnistamaan kriittisiä viivästyksiä projekteissa?
14. Miten näihin indikaattoreihin reagoidaan?
15. Miten eri tiimien ja osastojen välistä yhteistyötä hallitaan projektien aikana?
16. Miten projektit reflektoidaan ja mitä tapoja siihen kuuluu? Miten projektien opit dokumentoidaan ja siirretään tuleviin projekteihin?

Yleinen näkemys:

17. Mitkä ovat mielestäsi nykyisen projektihallintamenetelmän vahvuudet?
18. Entä haasteet?
19. Kuinka onnistuneesti nykyinen projektihallintamenetelmä mielestäsi toimii asteikolla 1-5? (1=erittäin heikosti, 2=melko heikosti, 3=kohtalaisesti, 4=melko onnistuneesti, 5=erittäin onnistuneesti)
20. Onko jotain, mitä haluaisit tuoda esille tai lisätä?