Karplund Elias

SCALING AGILE TO DELIVER CUSTOMER VALUE: MAPPING CHALLENGES AND ROLE OF CUSTOMER VALUE IN THE DEVELOPMENT MODEL OF A TELCO COMPANY

UNIVERSITY OF JYVÄSKYLÄ
FACULTY OF INFORMATION TECHNOLOGY
2023

ABSTRACT

Karplund, Elias
Scaling agile to deliver customer value: mapping challenges and role of customer value in the development model of a telco company
Jyväskylä: University of Jyväskylä, 2023, 111 pp.
Information Systems, Master's Thesis
Supervisors: Marttiin, Pentti & Suoraniemi, Kimmo

This thesis deals with scaled agile software development methods and their challenges to support customer value-based development decisions. Principles of agile software development from its manifesto have disrupted the software industry and are now mainstream. Its benefits have also attracted large and incumbent organisations to apply it to their operations, creating a need for scalable methodologies that seek to transform the original agile methodologies from their smaller contexts into those suitable for large organisations.

However, scaled agile methods such as SAFe come with their own challenges and their successful use requires significant efforts from organisations to avoid and mitigate them. These challenges can be related to the methods themselves, conflicts between different methods, cultural or domain-specific issues, and complexity. Through the concepts of agility companies are increasingly seeking to base their development efforts on customer value and to bring its impact into practice. Customer value has long attracted widespread interest particularly in marketing research but there is no unanimous definition of it nor standard methods for quantifying it, apart from customer experience measures such as NPS. Despite customers being on the forefront of agility, there practically are no established methods to incorporate it objectively to agile development.

The study was conducted with thematic interviews from its commissioning organisation which is an established telecom provider in Finland. It has been using a SAFe 5 framework-based development model for several years. Thematic analysis was used in the analysis of the data, from which themes for contributions and suggestions were recognised. The challenges identified in the research of scaled agile methods were found to apply in the target company, although the maturity of the model had shaped the current challenges from growing pains to more domain-specific direction. In terms of customer value, its measurement in the company was consistent with the few metrics in literature and it was not used as an objective metric in prioritisation. Concrete ways to increase the role of customer value in the company's development model were found to include more regular introduction of customer perspectives to first phases of analysis, a more value stream-oriented structure, and the implementation of a customer value component to the prioritisation mechanism.

Keywords: agile software development, scaling agile, customer value, measuring, prioritisation

TIIVISTELMÄ

Karplund, Elias
Asiakasarvoa ketteryyden skaalaamisella: haasteiden ja asiakasarvon roolin kartoitus teleyrityksen kehitysmallista
Jyväskylä: Jyväskylän yliopisto, 2023, 111 s.
Tietojärjestelmätiede, pro gradu -tutkielma
Ohjaajat: Marttiin, Pentti & Suoraniemi, Kimmo

Tämä pro gradu -tutkielma käsittelee skaalautuvia ketteriä kehitysmenetelmiä ja niiden haasteita asiakasarvoon perustuvan kehityksen tukemiseksi. Ketterä ohjelmistokehitys ja sen periaatteet manifestinsa myötä ovat muuttaneet ohjelmistoalaa merkittävästi ja ovat tänä päivänä valtavirtaa. Sen edut ovat saaneet myös suuret ja vakiintuneet yritykset kiinnostumaan sen soveltamisesta niiden toimintaan, mikä on luonut tarpeen skaalautuville menetelmille, jotka pyrkivät muovaamaan pienempiin konteksteihin tarkoitetuista alkuperäisistä ketteristä menetelmistä suuriin organisaatioihin soveltuvia.

Skaalautuvat metodit kuten SAFe eivät kuitenkaan ole pulmattomia ja niiden onnistunut käyttö edellyttää organisaatioilta suuria ponnisteluita haasteiden välttämiseksi ja lieventämiseksi. Nämä haasteet voivat liittyä muun muassa metodien käyttöön, konflikteihin eri metodien välillä, kulttuurillisiin tai toimialakohtaisiin erityispiirteisiin sekä monimutkaisuuteen. Ketteryyden konstein yritykset pyrkivät yhä enemmissä määrin perustamaan kehitystoimenpiteitään asiakasarvoon sekä tuomaan sen vaikutuksia käytäntöön. Asiakasarvoa on jo pitkään tarkasteltu markkinoinnin tutkimuksessa, mutta sillä ei ole yhtenäistä määritelmää eikä vakiintuneita metodeja sen määrälliseen mittaamiseen asiakaskokemusmittareita, kuten NPS:ää lukuun ottamatta. Metodeja sen objektiiviseen huomiointiin ketterässä kehityksessä ei myöskään käytännössä ole.

Tutkimus toteutettiin teemahaastatteluina vakiintuneelle Suomessa toimivalle teleoperaattorialan yritykselle, joka on jo useamman vuoden ajan hyödyntänyt SAFe 5-viitekehykseen pohjautuvaa mallia kehitystoiminnassaan. Aineiston analyysissa käytettiin teema-analyysia, jossa tunnistettiin teemoja tutkimuskontribuutioille ja parannusehdotuksille. Skaalautuvien kehitysmenetelmien tutkimuksessa todettujen haasteiden todettiin olevan laajalti läsnä myös kohdeyrityksessä, vaikkakin sen mallin vakiintumisen nähtiin muovanneen tämänhetkisiä haasteita kasvukivuista toimialakohtaisempiin piirteisiin. Asiakasarvon mittaaminen kohdeorganisaatiossa vastasi sen vähyyttä tutkimuksessa ja se ei ollut suoraan käytössä objektiivisena mittarina priorisoinnissa. Konkreettisiksi keinoiksi asiakasarvon roolin kasvattamiseksi yrityksen kehitystoimissa löydettiin muun muassa asiakasnäkökulmien säännöllisempi esilletuonti alkuvaiheen analyysiin, arvovirtakohtaisempi rakenne ja asiakasarvoa heijastavan mittarin käyttöönotto kehityskohteiden priorisointimekanismiin.

Avainsanat: ketterä ohjelmistokehitys, ketteryyden skaalaaminen, asiakasarvo, mittaaminen, priorisointi

FIGURES

TABLES

TABLE OF CONTENTS

1 INTRODUCTION

In today's business environments, companies are increasingly interested in the benefits of agility. In fact, becoming agile can even be seen as a necessity to maintain one's position in today's turbulent markets, where disruptors effortlessly leverage new technologies through their natural form, which incumbents struggle with. (Ebert & Paasivaara, 2017) Many see agility as a remedy to this challenge, but becoming an agile, established player does not happen overnight. Many challenges lie in the way of transforming existing operations to agile ways, and there are several methods available to guide companies through their transformation process. Despite their flashy pitches, these methods still do not guarantee success and emphasise responsibility of the company to do the heavy lifting to gain benefits (Kalenda et al., 2018). This first chapter discusses the main elements of this thesis, such as its framing, intentions, structure, and the research process and methodology.

1.1 Research setting

The purpose of this thesis is to study scaled agile frameworks and their challenges in large companies alongside a particular focus on customer value. Challenges of scaling agile are identified from existing literature and compared with those that emerge from the interviews with employees of the commissioning company of this thesis. The role of customer value in agility and how development decisions such as prioritisation could be based on it are also investigated in order to present suggestions for the company to improve its scaled agile development model. In other words, the aim is to investigate how customer outcome, rather than just the amount of development output, could be quantified in a numerical form so that prioritisation decisions and development as a whole could be based on it more objectively.

Agility and its methods have attracted a great deal of research attention, where the more recent and growing interest to scaling it forms its own research

stream (Edison et al., 2022). The contributions of this thesis are essentially two-fold:

1. Providing new and relevant information to the research stream of scaled agile methods
2. Providing information and practical suggestions for the target company to improve its implementation of the most popular scaled agile method, the Scaled Agile Framework (SAFe) (*16th State of Agile Report*, 2022)

In addition to these two, the focus on exploring customer value as a prioritisation factor intends to yield new information for the cross section of customer value and (scaled) agile development research, which has not attracted notable interest to date. This is despite of customers being at the core of agility in its manifesto (Beck et al., 2001), which serves as the backbone and philosophical foundation of agile software development.

Alignment between team autonomy and planning creates perhaps the most central issue of scaling agility successfully. Autonomy of experts to make the best decisions in teams without the constraints of management is central to truly agile development and drives its benefits (Moe et al., 2021). At the other end of this alignment spectrum lies planning, which is used as a tool to manage the complexity that large organisations often struggle with (Klakegg et al., 2016). The contradictory nature of balancing this autonomy with planning makes it difficult for organisations but simultaneously potential to explore in research.

Companies such as the target of this thesis are increasingly aiming to realise the buzz of customer centricity and deliver value to customers, which highlights the timely relevance of this topic. The practical context of this thesis is a major telecommunications provider in Finland with over 3000 employees. It has long roots in the Finnish market and currently uses a SAFe 5-based model as its way of working. It plays a key role in organising development in the organisation by combining essential stakeholders from different departments to collaborate effortlessly and efficiently – at least in the presentation pitches of SAFe. In practice, views of the current strongly SAFe-based way of working and even SAFe itself have often not been described with these words (*The SAFe Delusion – Information for Decision-Makers Considering the SAFe Framework*, 2022).

During a period of employment in the company's development function in Finland, I encountered shortcomings and commonly discussed challenges in the current SAFe configuration. This criticism raised my interest to study scaling agility. Practice-based remarks were formulated to research questions through discussions with the most important stakeholders of this thesis: academic supervisor from the university, instructor of practice with strong experience of agile from the company, and my previous manager. This has ensured balanced viewpoints and guidance from the initial drafting of the topic to drawing findings from the empirical material, all while ensuring the levels of confidentiality and openness the company and university require.

The study begins with a literature review which is followed by empirical methodology. The interviews were conducted in semi-structured style with interviewees carefully scoped around the company's development model. Thematic analysis was used to identify common themes from the transcribed interview material. The later findings and suggestions for the target organisation are based on this content and viewed in the light of the literature review in the discussion section. The results of this study are expected to be twofold: firstly, the correspondence of the identified challenges in scaling agility between the target organisation and literature. Secondly, shedding light on customer value's role and promotion in agile development decision-making such prioritisation. These remarks lead to the following research questions:

RQ1: What challenges are there in scaling agile and its methods?

> **RQ1.1:** Do the challenges in the target organisation correspond to ones found from literature?

RQ2: What models and methods are there to measure customer value and use it as means of prioritising agile development?

> **RQ2.1:** How does the target organisation promote the role of customer value in development prioritisation?
>
> **RQ2.2:** What means are there to increase this?

The literature review process of this thesis largely follows the three-step method created by Levy & Ellis (2006). They propose an effective information systems literature review to consist of these stages:

1. Inputs
2. Processing
3. Outputs

The first inputs stage involves collecting and evaluating potential works for use, after which they are processed in the second stage. Finally in the third phase, outputs are produced by writing the review itself. They target their process specifically at information systems due to the diverse nature of the literature but add that their method is very applicable to other areas of research as well. This supports its selection for this thesis as its topics span across agile, scaled agile, and customer value research, the latter of which differs from the first two.

The literature referenced in this thesis is of high quality and has been collected by using Semantic Scholar, Jykdok and Google Scholar scientific search engines. Special attention has been paid to the source of the chosen articles, which come from reputable peer-reviewed academic publications and industry sources. Search terms which were used in finding these articles include agile software development, scaling agile, method, SAFe, customer prioritisation, customer value, and their combinations. Many articles were left out of the thesis if the quality of their publication could not be verified using a Finnish publisher

ranking tool Julkaisufoorumi or other means. The main research streams of the explored literature were (scaled) agile development, challenges of scaling agile, and customer value and experience research. An open-source reference management software, Zotero, was used to store and manage the works used in this thesis and to enforce formatting guidelines.

1.2 Structure

The literature review progresses in a chronological order. It starts with describing the roots of the agile paradigm shift in software development, which reach as far as the 1930s (Larman & Basili, 2003). In 2001, a manifesto signed by influential members of the software development industry was published as a call for practitioners to move from heavy and rigid, plan-driven methods towards developing software in more flexible and human-centric, agile fashion (Beck et al., 2001). Based on the principles of this manifesto, two of the most well-known methods in the software industry, Scrum and Extreme programming (XP), are explained through their ideas on how software development should be conducted. Principles of Lean software development are gone through as they are often discussed together and complement each other with their guidelines.

Onwards from the spark and spread of agile and its original methods, the subject moves on to scaling it. Modern, large organisation are increasingly interested in gaining the benefits smaller actors have harnessed from agile and lean ways (Ebert & Paasivaara, 2017). However, implementing them to large and complex environments requires special recipes, where scaled agile methods come to play (Laanti, 2014). The fundamentals of scaling agile are explained, followed by describing the most used method, the Scaled Agile Framework (SAFe) how it empowers companies to embark on journey of agile transformation to maintain and increase competitiveness in modern, turbulent markets with its principals and competencies (Knaster & Leffingwell, 2020, pp. 24–25).

Despite of existing methods, scaling agility is difficult to perform successfully. There are many challenges recognised in research when it comes to transforming operations to agility that root from various causes. (Conboy & Carroll, 2019; Kalenda et al., 2018) A common high-level theme for these challenges is the fundamental issue between autonomy of agility and forward planning that companies still necessitate to deal with their complexity. A key challenge for an agile transformation is to find out where the optimal alignment between these two exists. (Moe et al., 2021)

Despite customers having a profound status in agile methods, measuring it is fundamentally difficult (Kreuzer et al., 2020). This challenge is the main topic of the fourth chapter of this thesis, where customer value and measuring are first defined and then viewed in the context of agile development methods and how some prioritisation mechanisms incorporate customer value to them. Measuring customer value is challenging due to there being no agreed upon meaning of the term customer value itself (Gallarza et al., 2011). Additionally,

the term's sentimentality with the lack of numerically measurable attributes further hinders its objective measurability (Kreuzer et al., 2020). Despite the fundamental difficulty in its direct measuring, subcomponents of customer value, such as customer experience can be attributed numerically with proven metrics such as NPS (Reichheld & Markey, 2011).

The fifth chapter is dedicated to summarising and discussing the literature review presented through earlier chapters. The summary and discussion view the subjects in a bigger picture and present key findings as basis for the following empirical study. The sixth main chapter details the empirical methodology that was first used to gather the data through semi-structured interviews and its analysis with thematic analysis method. Also, the nature and scoping of interviews are explained. The seventh chapter describes the target company in more detail to increase understanding of the empirical context and provides information of the personnel who were interviewed.

The findings of the empiric study are presented in chapter eight, where four main themes emerged of the challenges in the target company's model. Interviewees' takes on customer value as a term as well as its nature and its promotion in the current model and the company's domain in general are some of the integral contributions the empiric findings provide together with the characteristics of the observed challenges. The following ninth chapter discusses the connections between the empirical findings and literature of the earlier review to spot similarities and differences. These are further expanded upon in the tenth chapter, which provides conclusive summarisations with detail on the practical and academical contributions of this work alongside its limitations and potential directions for future research.

2 FUNDAMENTALS OF AGILE

In the core of this thesis is the agile paradigm shift in software development. This section is dedicated to explaining the birth and spread of this phenomenon. The core principles which differentiate agile from previous ways are explained providing necessary background information to understand their significance. Agile methods have even inspired other domains to adopt their key principles, which is telling of their success (Hohl et al., 2018). Particular attention is paid to three essential pillars of agile: customer, iterativity, and incrementality. While there are many agile methods proposed in research and practice, this thesis is scoped to include two of most used ones to use them as exemplary methods: Scrum and Extreme programming (*16th State of Agile Report*, 2022).

2.1 The agile manifesto

In 2001, a group of software industry pioneers came together to synthesise new ways for software development, which had already started to take shape in the industry in various forms. These seventeen experts, who even labelled themselves as "organisational anarchists" signed the Agile Manifesto, whose 12 principles aimed to improve ways of working in all sides of development process. This was to be done by moving the focus from formal, strict, and heavy plan-driven processes to more human, practical, collaborative, and flexible habits. Focus on formal concepts and interests such as tools, contracts and planning were not meant to stop, but to be set aside from the main focus points. (Beck et al., 2001). The core statement of the Agile Manifesto is presented below:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

From the core statement, it is apparent, that it does not provide practices but instead lights the flame to change fundamental principles of the industry. The 12 principles have more insight to the core message focusing on more specific aspects. The first principle by Kent Beck et al., (2001) can be viewed as the summarisation of the goal and required interactions of the agile spirit: "Our highest priority is to satisfy the customer through early and continuous delivery of valuable software".

Setting the customer as the first principle marks its significance in the agile paradigm. Compared to previous plan-driven methods, Abrahamsson et al. (2002) summarise how truly agile development occurs when both customers and developers collaborate with tight communication. But to deliver value from completed work, it must also be delivered and according to this principle, "continuously." The concept of continuous delivery is closely connected to the essence how work is structured in agile development: iterativity and incremental development (Larman & Basili, 2003). Iterativity in agile software development encompasses the plausibility of being wrong by laying activities to short cycles instead of long and rigid phases. These cycles can be repeated to build on previous results to get to a desirable goal, which often could not be achieved in a single development phase due to lack of refinement. Incrementality goes hand in hand with this, as it characterises developing systems in incremental pieces instead of one. These building blocks can be created and tested simultaneously then implemented to the system. (Miller, 2001)

Iterativity and incrementality have existed in the industry long before the manifesto. Larman and Basili (2003) trace elements of them to projects at IBM as early as in the mid-1950s. Still, it was not until the agile manifesto that incrementality started attracting true attention in the industry, despite of the inferiority of waterfall methods being acknowledged at least in research much earlier (Larman & Basili, 2003). In fact, the concepts of agile can be viewed simply as a collection of best practices and their evolution in the industry by uniting their previous scatteredness (Miller, 2001).

2.2 Agile software development methods

Abrahamsson et al., (2002) describe a method to contain the practices and work products a process should use. Even before the Agile manifesto, development methods which can be classified as agile surfaced in the software industry. Some of the most well-known agile methods since their inception are Scrum and Extreme Programming (XP) (Anwer et al., 2017). Despite of agile methods having common ground with the manifesto's principals, many practitioners do not understand that they all have their unique characteristics (Hohl et al., 2018). Still, software organisations following Scrum or XP agile development methods reported more satisfactory results than their plan-driven counterparts (Salo & Abrahamsson, 2008).

In this subsection Scrum and XP are presented with their key fundamentals, such as responsibilities, values, and processes. They both surfaced around the time of the agile manifesto but have history from earlier years and even decades (Beck, 1999b; Schwaber, 2004). They shaped to their known form with inspiration of concepts that nowadays are considered agile, namely iterativity and incrementality. These can be traced back to as early as 1930s, when potential of feedback collection from cyclical project structure was noticed. (Larman & Basili, 2003) Next, based on the presented principals and fundamentals two agile software development methods are discussed in more detail.

2.2.1 Scrum

The first appearance of the term Scrum was in 1986, when two Japanese experts described how a formation from rugby could be reflected to product development. Instead of passing the ball (product) to the next responsible in line, they proposed that better results could be achieved if all players (experts) traversed the journey to the objective with the ball together. In the process players, experts of their own arts would chip into the process autonomously when needed by working closely with the product and each other throughout the whole development journey. This would increase speed and flexibility of the process which would meet modern market demands. (Takeuchi & Nonaka, 1986)

Inspired by Toyota's success with transforming management culture and philosophies, the cultural essence of Scrum boils down to allowing independence and promoting self-direction of developers as complexity increases. It aims to shorten feedback loops between customers and developers to grasp changes in time with trial and error being acceptable ways of working and outcomes in the process. As the result, value of the developed items is constantly tested, inspected, and built upon – forming a so-called business feedback loop. Scrum intends teams and their members to be in charge of their own fate by bringing decision making from bosses to the boots on the ground, which should yield new angles to solving problems (Schwaber, 2004, pp. xii–xv).

As a natural consequence to independence and freedom comes responsibility. Ken Schwaber introduces Scrum as "deceivingly simple" as an agile foundation, which does not provide precise instructions but instead aims to establish an environment and processes where teams can succeed based on their own competences and decision making (Schwaber, 2004, p. xvii). Scrum is targeted towards managing projects more transparently with the customer on the forefront (Schwaber & Beedle, 2002). Scrum has been designed to be ready for changes in the development, such as requirements, from the ground up, something what traditional methods struggled with (Abrahamsson et al., 2002).

The structural core of Scrum is based on the concept of iterations. Results of the previous phases will be analysed to improve on while accommodating emerging changes and challenges continuously during an iteration (Schwaber, 2004, p. 17). In practice, iterations in Scrum are called sprints which target to deliver the maximum amount of quality software and therefore value in under 30 days (Beedle et al., 1998). The worked items of each sprint are chosen from a

Product backlog, which is a prioritised list of functionalities, features, and technology. The product backlog is never complete, evolving during the product's lifecycle with new items appearing from any product stakeholder. Each sprint also has its own, more specific backlog, a sprint backlog (Schwaber & Beedle, 2002, pp. 7–9).

Scrum bases itself on five values which its participants exhibit: commitment, focus, openness, respect, and courage (Schwaber & Beedle, 2002, p. 147). Scrum is built around three roles: the product owner, the team, and the Scrum master. The product owner is interested in the outcome of the project and manages requirements which are transformed to items on the product backlog. The backlog is a storage of development items which the product owner prioritises by the value items will create once ready. Project funding, return on investment calculations and release plans are also product owners responsibilities (Schwaber, 2004, p. 19).

The team employs the freedom and autonomy mentioned earlier to turn items on the product backlog to reality and therefore, value. In other words, the team delivers functionalities based on the product owner's vision but with their own view and responsibility on the practical development, granted by their expertise. Therefore, the team members manage the iteration how they see is the best to produce as much value as possible. (Schwaber, 2004, p. 19)

Finally, the Scrum master can be seen as the coach who ensures that the development follows the Scrum process and procedures. They make sure that all members and even organisation implement Scrum to its full potential in their practices and provides guidance when necessary (Schwaber, 2004, p. 19). To think of Scrum from a management perspective, it is apparent that as a method it brings management responsibilities closer or in many cases directly to boots on the ground. This is particularly present in the case of the team, which has the responsibility and freedom to make their own decisions regarding development items based on their expertise. The role of the Scrum master is also different from a typical manager who instead of micromanaging development procedures is employed with a coach-like mentality to harness everyone with the best practices. While all this may sound great, in practice even Schwaber himself describes that for managers, finding the management balance in scrum is very difficult (Schwaber, 2004, p. 28).

The roles of Scrum come to life in its structure, where the Scrum flow arranges them followingly: In the beginning of a sprint a planning session is held, where the product owner and the team discuss content of the product backlog and details of items which should be picked for the next sprint. The role of the team is to present questions to the product owner and determine what can realistically be achieved in the sprint. The team selects items from the value-prioritised backlog which it has confidence in turning to a completed increment during the sprint. After presenting the selected items to the product owner, the team spends time to build a sprint backlog where the selected product backlog items are divided to smaller tasks for the team to complete. The progress of these tasks is reviewed in daily scrums, 15-minute meetings in the beginning of

each day. In the end of the sprint, it is closed with sprint review session, where the team presents sprint's results for the product owner and other interested stakeholders. After sprint review, the scrum master gets together with the team for a sprint retrospective to discuss the sprint and improve practices for the next one. (Schwaber, 2004, p. 20) Scrum's process flow is visualised in Figure 1.

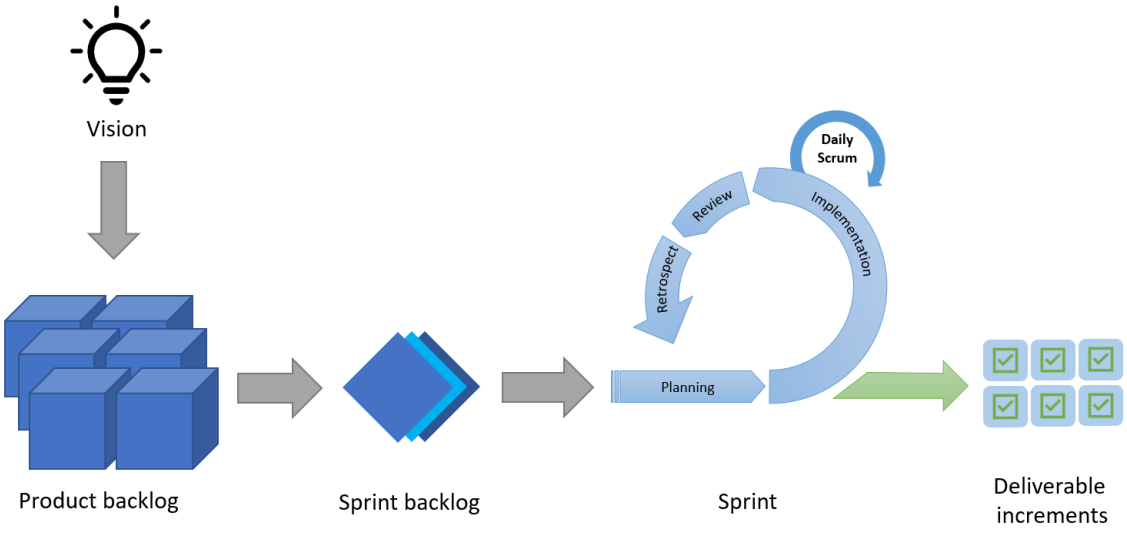


Figure 1 Flow of Scrum presented through its phases and artifacts (Schwaber 2004, p. 24)

2.2.2 Extreme programming (XP)

Extreme programming shares many fundamentals with Scrum but is still a distinguished agile software development method of its own. Its agile roots lie in incrementality and iterativity, and as a method it was first officially published by Kent Beck in 1999. (Beck, 1999b) XP underlines values of communication, feedback, simplicity, courage, and respect. These values are expressed with practices and translated to value with principles, if a practice can't be applied directly to a problem (Beck & Andres, 2004, p. 33). Like Scrum, XP as well takes inspiration from product development ideas of Takeuchi and Nonaka (1986) to increase speed which modern market conditions require.

In its structure Extreme programming transforms familiar phases from plan-driven methods to as small increments as possible which enables collecting feedback in the spirit of iterativity and incrementality. For example, instead of analysis and design having phases of their own, they are processed within each small increment. This makes XP more responsive to changes. Extreme programming contains 12 practices: The planning game, small releases, metaphor, simple design, testing, refactoring, pair programming, collective ownership, continuous integration, 40-hour week, on-site customer, and coding standards. (Beck, 1999a)

The planning game practice is where releases are scoped. The result is created by balancing between business demands (scope, priority, composition of releases) and technical details (estimates, consequences, process, detailed

scheduling) where the plan should be always updated as soon as possible should realities change. This practice is one of the key details of XP. Small releases, as its name indicates, steer development to push new versions on a rapid pace. Metaphor empowers the development with an easily comprehensible and shareable story of the system's functioning. Simple design highlights how complexity should be strictly avoided and if found, removed at once. (Beck, 1999b, p. 46) Development should only continue if unit tests pass without an error. Also, to indicate that developed features are finished, the customer should also create tests. Refactoring, improving quality without affecting the system's behaviour is also an encouraged practice. Pair programming emphasises that two coders should write all code in production on a single computer. Collective ownership allows anybody to alter all the system's code anytime. Continuous integration instructs to build and integrate the system always when a task is ready. 40-hour week limits working hours and forbids overtime for a second consecutive week. On-site customer marks the importance of having the customer ready for questions always when needed. Coding standards enforce standards, such as a communication through code. (Beck & Andres, 2004, p. 47)

XP lays its optimal lifecycle to exploration, planning, release planning, iteration planning, productionising, maintenance, and death phases (Beck, 1999a, p. 99). An XP process starts with exploration, where the requirements and technological realities are mapped and defined. The customer is incorporated to the phase directly by having them create user stories (features), which act as requirements without technical details to provide insight to developers to understand customer wishes. Next in planning phase, a schedule for delivering the maximum value with selected stories is agreed between the customer and developers. The stories are chewed to iterations, which last one to four weeks. The customer decides which stories are picked to iterations forming test cases. (Beck, 1999b, pp. 99–101)

In productionising, iterations become shorter to only one week to boost the feedback cycle. In this phase, big design decisions are getting locked in place, and emphasis moves towards tuning for production deployment. (Beck, 1999a, pp. 101–102) After this, the maintenance phase focuses on upkeep of the developed system and implementation of possible new features. It highlights carefulness when operating a live system, learning from providing support for it, and transferring knowledge if there are people exiting or joining the project. (Beck, 1999a, pp. 102–103) In its final phase, the death phase, the customer stops providing new stories to implement to the system and XP recommends creating a document to describe some of key workings of the system. In reality, this often is the point where the customer is no longer satisfied with the system and features can't be economically implemented to meet the customer's demands (Beck, 1999a, p. 103).

Where XP incorporates the software project's whole lifecycle to its concepts, Scrum's focus lies more on the development process and project management. In fact in literature as well Scrum has been viewed as having more of an "man-

agerial" viewpoint and XP in turn a more "engineering" oriented one (Anwer et al., 2017). In other words, XP has a more hands-on view of the software process and proposes practices that can be more easily implemented, such as pair programming. Scrum is more centred to philosophies how it can steer development to function the best through learning and team autonomy. Perhaps this makes Scrum adoptable to a wider range of practical environments and has therefore played a part in its popularity today.

Iterativity and incrementality are in the heart of both methods, but XP provides hands-on steps how they enable an agile development cycle with its practices. On the other hand, this level of detail in practices has also generated critique, as they have been deemed incompatible to certain contexts (Petersen & Wohlin, 2009). Extending this line of thought even further, does true agility, based in the best decisions of tightly co-operating experts, flourish in a specifically instructed environment? Then again, each adopting organisation, and in agile spirit autonomous agile teams, should choose the practices they see the most fit and fruitful for themselves.

The principles agile methods base themselves on may seem obvious to those used to modern leadership guidelines. Dingsøyr et al. (2012) call agile practices "intuitive" as they have basis on widely proven management guidelines. Perhaps this summarises the even more high-level principle of agile: management with intuition how to treat other humans to create a nurturing atmosphere for innovation? Merisalo-Rantanen, Tuunanen and Rossi (2005) take the autonomy and self-directing ideals of agile methods a step further by declaring that focusing on utilising a specific method is irrelevant because motivated developers will be productive as long as a suitable environment without micromanagement is established, regardless of the employed development method. Perhaps instead of being too keen on best practices and their implementation, focus should be paid to applying agile values to their organisational culture to create a nurturing environment for experts and so increase quality and productivity.

2.3 Lean software development

Lean is a collection of philosophies to steer an environment towards effectivity and agility. Its origins root from the Japanese automotive industry, which struggled with producing cars in smaller amounts economically in 1930s which created the need for new production methods and concepts. (Poppendieck & Poppendieck, 2003, p. 9) Since then, Lean principles have evolved and received wide praise particularly in manufacturing industry, where it first surfaced as a term in the 1980s (Holweg, 2007). Its principles have also been proven effective in other industries, like software development, despite vast differences in their nature (Ebert et al., 2012). In 2003, Poppendieck and Poppendieck adapted Lean principles from manufacturing industries to software development by present-

ing seven key principles, and 22 more specific tools. In this subsection, the seven principles are explained with their core lines of thought.

Ken Schwaber calls Lean software development a framework, which consists of tools that reveal why and how agile processes operate and how they can be thoroughly understood to modify or build new agile processes (Poppendieck & Poppendieck, 2003, p. 6). Many similarities between previous agile principles and Lean principles can he spotted with even some overlap. Ebert et al. (2012) describe that Lean can help to extend the scope of agile methods from team/project level towards more sustainable long term decision making, such as evaluating lifecycle costs. Building on this we can think that lean should complement agile practices particularly in company environments, where managing complexity also necessitates long-term activities, such as planning. Poppendieck & Poppendieck (2003) list the principles of lean software development to consist of the following:

1. Eliminate waste
2. Amplify learning
3. Decide as late as possible
4. Deliver as fast as possible
5. Empower the team
6. Build integrity in
7. See the whole

The first solution to Toyota's inefficiencies was to eliminate waste, which formulates the first lean principle. Waste in the mind of the father of Toyota's production system, Taiichi Ohno, was "everything that does not create value for the customer." For example, he considered that waste from waiting could be eliminated by building cars directly from order to delivery instead of building inventory. Value is delivered to the customer when they receive the car, which eliminates waste from waiting. (Poppendieck & Poppendieck, 2003, p. 13)

In lean, eliminating waste is the most important principle which makes learning to spot it crucial. Its biggest sources should be identified and eliminated after which the same process is repeated to the largest remaining sources. This cycle continues to the point where even seemingly necessary sources of waste can slowly be removed. (Poppendieck & Poppendieck, 2003, p. 13)

The second principle of Lean is amplifying learning. Software development is not manufacturing, which renders best practices from Lean's original domain unideal. Lean's authors compare the process of creating a recipe to the software development process, whereas in manufacturing the goal is to follow a recipe as consistently as possible. Amplifying learning empowers the idea of trial and error to developing software: what can be learned from variations to produce the best quality to as many customers as possible? Designing solutions to complicated problems benefits from having different experts in the process without a set approach to finding a solution. For well-defined problems, a more set-in-stone process without experimentation may work fine but does not leave room for learning. For ambiguous problems trying, testing, and fixing may cre-

ate more value, as iterative processes often do. When the knowledge of the problem increases, preferably by business people working closely with the project, refactoring rounds can work as learning cycles for the final solution, therefore increasing its potential value. (Poppendieck & Poppendieck, 2003, p. 18)

Decide as late as possible is the third principle of lean software development. It is achieved with concurrent development, which means that production should not start only after design has seemingly been finished, but to perform these almost hand in hand. The trick is to not go in to too much detail in developing the solution by leaving space for more specific design decisions which will eventually follow. By developing and designing concurrently, important decisions can be made later which increases the knowledge that can be acquired from developing the solution thus far allowing room for changes. In software development this principle is usually applied by utilising iterativity and beginning work from the most valuable features when initial designs are set. This serves as an exploratory method to study different approaches before a stricter path is defined which limits options. Delaying decision making with concurrent development reduces one of the biggest sources of waste: changing software when it has already been developed which is usually very difficult and therefore costly. (Poppendieck & Poppendieck, 2003, p. 25)

Lean suggests that decisions should be made as late as possible, but the contrary is proposed for delivery. The fourth principle, deliver as fast as possible, is grounded in customer's love for speedy delivery. When customer receives their order fast the risk of changing requirements mid-production is also reduced by leaving less time for them to surface. This principle also encompasses the idea of shortening the value stream by avoiding completed, but not delivered work by, in the case of software, implementing continuously. Fast delivery also enables late decision making when changes can be implemented on short notice without strict forward planning. (Poppendieck & Poppendieck, 2003, p. 31)

The fifth Lean principle's history starts once again from the manufacturing industry. In the 1980s it became clear that legacy practices of scientific management were not effective and the move towards improving the working conditions and as the result, quality of the product initiated. The principle of empowering the team targets to get the most out of the experts who work around the product to make it the best. As key tools to achieve this, self-determination, motivation, leadership, and expertise suggested. (Poppendieck & Poppendieck, 2003, p. 36)

Build integrity in is the sixth principle of Lean. It comprises of two types of integrities: perceived and conceptual integrity. Perceived integrity means that the subject who uses the product finds it well balanced regarding function, usability, reliability, and economy: excellent and pleasing to use in other words. Perceived integrity requires its pair, conceptual integrity. It means that the concepts the system has been built around work smoothly in cohesion. Conceptual integrity is hindered, if there are for example parallel systems for a sole use case but fulfilling varying concepts. Conceptual integrity therefore also means archi-

tectural decisions which hide the gaps from users. (Poppendieck & Poppendieck, 2003, p. 44)

Systems thinking, the act of grasping interconnections between organisation's functions and how they operate, is the core of the seventh Lean principle: See the whole. It promotes to investigate the real consequences of actions which are performed to reveal how they worked or if they caused the observed impact in the first place. How complicated systems interact and adapt to decisions, such as where there are blockades to growth, are important topics for consideration. (Poppendieck & Poppendieck, 2003, p. 51)

The last principle Poppendieck and Poppendieck (2003) present in their book of Lean software development boils down to implementers to figure out the proper balance between the principles and practice. Instructions and Warranty – the name of the eighth principle also serves as a word of caution to not get too excited about Lean by invoking undesired consequences from applying the principles blindly. They state fittingly: "One team's prescription is another team's poison". (Poppendieck & Poppendieck, 2003, p. 56)

3 AGILE AT SCALE

Agile methods, such as Scrum and XP are targeted to small teams with around ten members with a concise amount of supporting key stakeholders in the development environment (Beck, 1999a; Schwaber & Beedle, 2002). These types of contexts are where agile practices were first inspired, created, and proven effective. In the aspirations to harness the much-hyped benefits of agile, traditional organisations, such as large companies or public organisations, have also become interested in agile ways of working. (Dingsøyr et al., 2018)

In research the interest towards scaling agile started gaining ground around a few years after the agile manifesto, when was noticed that organisations began applying agile methods to larger contexts despite the several drawbacks their application to these unintended conditions caused (Petersen & Wohlin, 2009). Large organisations and system development have distinguished characteristics and pure needs which make applying team-level agile methods fundamentally challenging (Boehm & Turner, 2005; Reifer et al., 2003). This means, that to apply agile to large company contexts, it has to be scaled properly, doing which is discouraged with original agile methods due to their misfit (Reifer et al., 2003). In the following parts of this thesis, agile methods targeted to small teams, such as Scrum and XP, are called original or traditional agile methods due to their earlier appearance in the industry to distinguish them from more recent scaled methods.

As an organisation grows larger, it becomes more difficult for it to change. This makes it harder to implement agility (Livermore, 2008), which instead of relying on individual functions to perform their sole role, roots from benefits of collaborative responsibility (Takeuchi & Nonaka, 1986). Agility of large organisations is fundamentally challenged by their complexity from increased interconnectivity between many internal functions. The dependencies created by these connections form rigidity that requires documentation to handle, contradicting with the spirit of agility (Lindvall et al., 2004). Companies should also accept the increased uncertainty, which is natural to agile methods due to incrementality where results should and cannot be precisely planned or forecasted (Boehm & Turner, 2005). Implementing agility successfully requires a pro-

cess called agile transformation, which replaces old ways of doing things with agile processes. Above all, this requires a radical change to the organisation's culture to succeed (Misra et al., 2010). Fitzgerald et al., (2014) describe digital transformation as "the use new digital technologies to enable major business improvements, streamlining operations or creating new business models". This can be seen as a comprehensive, high-level definition of the phenomenon.

Companies are increasingly interested in applying scaled agile to their operations to receive two-fold impacts: gaining benefits to its current state or solving present challenges (Uludağ et al., 2021). Agility is often first tried in small development teams, where it fits naturally. Extending agility from development activities is called large-scale agile development. This employs a definition of at least six teams and 50 or plus employees as in large organisations, multiple teams working in projects is not uncommon (Paasivaara et al., 2018).

Ebert and Paasivaara (2017) list six different aspects on which combining lean and agile practices as a framework can help in answering to business needs: business model, governance, process, competences, development, architecture. Uludağ et al. (2021) identified and listed the main motivators towards methods to scale agile effectively to be:

- Improving the agility/adaptability of the organisation
- Improving the collaboration of agile teams working on same product
- Improving the coordination of agile teams working
- Improving the synchronisation of agile teams working on same product

From this listing we can notice how the motivators behind scaling agile largely reflect the ideas presented in the Agile manifesto back in 2001. This is of course only natural, as scaling agile aims to reap the benefits original agile methods, based in the philosophies presented in the manifesto, are known for. The scaled motivators still have a profound interest in timeliness through coordination and collaboration, which understandably difficult to manage in a large environment as there are more moving pieces which need to collaborate. The ambitiousness together with the team autonomy from agile transformation can scare its embracers, or oppositely, attract them to set it in motion with the velocity it requires.

3.1 Scaled agile methods

To effectively employ and harness agility to large organisations, methods and frameworks have been created by experts in agility-seeking companies and established members of the agile community (Uludağ et al., 2021). Laanti (2014) describes scaled agile methods to extend agility, previously guiding individual software development teams, to empower the whole organisation. There are numerous original agile methods and the same applies to scaling agile to large

environments, which have increased in popularity and amount after the boom of original agile methods. The overwhelming majority of frameworks have been published between 2011 and 2018 and it is not unusual for them to have similar elements as many authors have taken inspiration from one another. Frameworks also often have multiple versions with newer ones refining and adding content. (Uludağ et al., 2021) In research, large-scale agile as a phenomenon and its methods have attracted more and more attention in recent years, which is telling of increased interest towards them (Edison et al., 2022).

The progress from the first appearances of agile concepts to its scaling methods today is presented in Figure 2. The exact dates for first appearances of some methods and concepts are difficult to assign to a specific year, so dates of first major publications of methods are used as their birthdates. Blue boxes in the figure present concepts and events while green boxes are used for methods and frameworks discussed in this thesis.

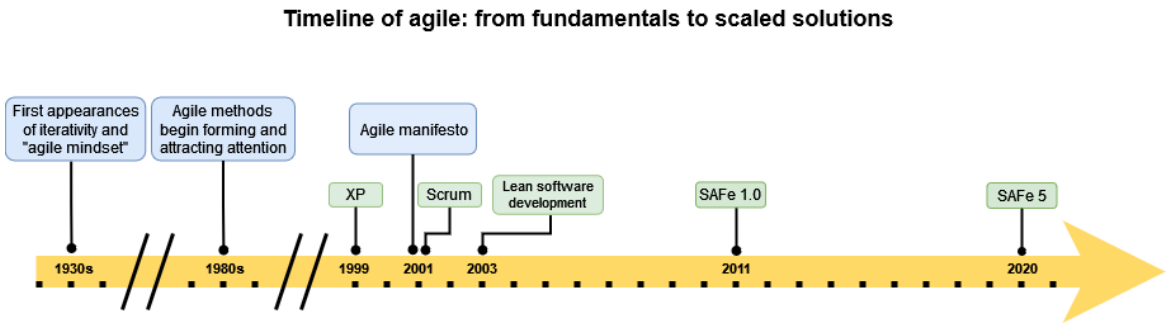


Figure 2 Timeline of agile: from fundamentals to scaled solutions (Hohl et al., 2018; Larman & Basili, 2003; Uludağ et al., 2021)

Alqudah and Razali (2016) propose the most prominent scaled frameworks and methods to consist of Disciplined Agile Delivery (DAD), Large Scale Scrum (LeSS), Scaled Agile Framework (SAFe), Spotify, Nexus, and Recipes for Agile Governance in the Enterprise (RAGE). A more recent study by Edison et al., (2022) concluded on almost identical results, where SAFe, DAD, Spotify, and LeSS were the most popular methods in the responses from the industry. Additionally, they found how another popular approach has been to create an own method by scaling an original agile method from team level. This practice has not decreased in popularity possibly due to organisations preferring the ease advertised by commercial solutions, such as SAFe, despite them also requiring customisation and implementation effort. (Edison et al., 2022)

To maintain a concise scope, particular attention is paid to the Scaled Agile Framework (SAFe) as it is by far the most popular in the industry (*16th State of Agile Report*, 2022) and is the one implemented by the target organisation in the empirical part of this thesis. By providing more details of SAFe, its configuration and operation in the target organisation can be studied with more information.

3.2 Scaled Agile Framework (SAFe)

The Scaled Agile Framework (SAFe) is a package of empowerment for large companies to transform and become agile. Its purpose is to transform large companies to deliver better solutions faster than their rivals not only regarding software development, but by applying the best principles, practices, and guidance comprehensively to all its operations. This will create a process called "agile transformation": a radical shift from legacy ways to agility (Knaster & Leffingwell, 2020, pp. 23–24). Multiple versions of SAFe have been published since its debut in 2011. Its latest version at the time of writing is 5.1, which was released in 2021 ('Advanced Topic - What's New in the SAFe 5.1 Big Picture', 2021). The following descriptions are based on the official SAFe 5 Distilled book, where the framework's contents are described in detail with examples. Hence, it must be noted, that SAFe is a commercial solution with their material always having their own interest in mind. This section aims to deliver a balanced viewpoint to SAFe aside showy, even bold claims in its reference material.

The four guiding values of SAFe are alignment, built-in quality, transparency, and program execution (Knaster & Leffingwell, 2020, p. 116). The authors of SAFe, Richard Knaster and Dean Leffingwell (2020, p. 45) claim that transformation through implementation of SAFe will provide 35% boost in productivity, 50% increase in employee satisfaction, 50% increase in time-to-market speed, and 50% decrease in amount of defects. SAFe has also attracted a lot of controversy and critique from influential practitioners, such as creators of popular original methods and signers of the agile manifesto. Many call SAFe an easy-to-sell framework crafted by far-from-practice consultants , which despite its glorious claims, does not operate in agile fashion and fails to deliver benefits in action (Hobbs & Petit, 2017; *The SAFe Delusion – Information for Decision-Makers Considering the SAFe Framework*, 2022). Nevertheless, SAFe is the most popular method for scaling agile (*16th State of Agile Report*, 2022).

SAFe is intended for various environments from smaller ones with 50 to 125 people to even thousands (Knaster & Leffingwell, 2020, p. 45). It offers four configurations which organisations can choose from: Full SAFe, Portfolio SAFe, Large Solution SAFe and Essential SAFe. Appendix 1 contains the SAFe 5.1 Big picture-poster where its elements are laid out. Essential is the most barebones version with others providing additional features on top of it (Knaster & Leffingwell, 2020, p. 50). There is also a specific SAFe version tailored for adaptation to public sector (Knaster & Leffingwell, 2020, p. 54). As SAFe focuses on transforming organisations overall, its adoption relies on comprehensive embracement of agile ways in the whole organisation through processes, tools, and roles, rather than individual tools on team level (Edison et al., 2022). This means that referencing SAFe as a toolset may mislead its core mission and the fundamentals empowering it.

As SAFe truly is a massive framework, this thesis explains its most important guiding values and principles with its more practical competencies to

describe how SAFe steers companies to ultimately achieve agility at scale. Particular attention is paid to team and technical agility and lean portfolio management competencies as they contain specifics around SAFe's structure and managing backlog items, which is more relevant for the topic of this thesis.

3.2.1 Principles

SAFe's principles and practices are based on lean-agile mindset, which harnesses the two paradigms to work in harmony to empower transformation and its goals (Knaster & Leffingwell, 2020, p. 61). Understanding them is a precondition to apply SAFe's practices and roles, because the principles are not contextual whereas practices and roles are and may so face various challenges (Knaster & Leffingwell, 2020, p. 78). Based on the thought lines of these two, Knaster and Leffingwell (2020, pp. 78–79) present 10 principles that SAFe's roles and practices are based on. They are:

1. Take an economic view
2. Apply systems thinking
3. Assume variability; preserve options
4. Build incrementally with fast, integrated learning cycles
5. Base milestones on objective evaluation of working systems
6. Visualise and limit Work in Process (WIP), reduce batch sizes and manage queue lengths
7. Apply cadence; synchronise with cross-domain planning
8. Unlock motivation of knowledge workers
9. Decentralise decision-making
10. Organise around value

These principles are crucial, particularly for management to comprehend to foster a culture where SAFe's implementation can succeed because only then can SAFe provide its benefits (Knaster & Leffingwell, 2020, p. 107). The first principal targets development to create value as early and often as possible while accounting for economic realities. Its key concepts are incrementality and continuous delivery which together unleash value much earlier than their plan-driven counterparts. (Knaster & Leffingwell, 2020, p. 81) Applying systems thinking aims to tame complexity that the system and its builder organisation inflict upon it and to understand fundamental value streams by focusing on the big picture instead of individual system components (Knaster & Leffingwell, 2020, p. 83). Assuming variability and preserving options guides to always leave room for alterations by defaulting to possibility of changing plans (Knaster & Leffingwell, 2020, pp. 86–87). The fifth principle instructs to monitor progress with concrete results which indicate value during development rather than the result. To achieve this, SAFe distributes smaller portions of requirements, design, development, and testing activities to increments to build the solution piece by piece, allowing it to be evaluated during its development. (Knaster & Leffingwell, 2020, pp. 90–91)

The sixth principle of SAFe concentrates on managing the amount of Work in Progress development (WIP) and optimising lead times. Having too much WIP creates congestion and various other negative side effects without positives. To avoid this, SAFe guides to visualising work, for example by utilising Kanban boards which organise development items by their status to an easily comprehensible format. Reducing the size of batches and their transactional costs are also promoted as means to increase flow of work and therefore value delivery. Managing queue lengths is also an important activity to prevent prolonged delivery and waste from lean terminology. Optimal queue lengths also provide quality benefits and increase predictability. (Knaster & Leffingwell, 2020, pp. 92–95) SAFe's seventh principle highlights cadence and synchronisation through domains. The authors describe that the freedom where agility provides its benefits requires cadence and synchronisation to provide information of the present, which together manage uncertainty. Cadence is fundamentally linked to the previous principle because it can be used as a means of scheduling and arrangement. Synchronisation is more focused on coordination of a systemic development environment, where Program Increment (PI) planning events are central in establishing actions to execute the common plan. (Knaster & Leffingwell, 2020, pp. 95–97)

The eighth principle promotes the role of knowledge workers, their compensation, expertise and how it can be fully harnessed with proper lean-agile managerial support and culture. Once again, systemic thinking is seen as the right way to steer experts towards, alongside nurturing an engaging environment where experts can voice their opinions with influence. (Knaster & Leffingwell, 2020, pp. 99–101) The ninth principle also touches on the role of experts by explaining that decentralising decision making in beneficial to the flow of development and so, lead times. Strategic far-reaching decisions should still stay in the hands of leaders, but teams and their experts should be trusted with everything else. (Knaster & Leffingwell, 2020, pp. 101–102)

The last principle of SAFe is called "Organise around value". This statement is seen as the requirement to achieve business agility: "delivering more quickly". Companies should be able to perform adaptation around value quickly and without struggle always when new streams of value are observed. A portfolio in SAFe is a "collection of development value streams" where agile teams, Agile Release Trains (ARTs), and solution trains serve as the building blocks from small to large which are used to construct aligned value delivery around streams. All these building blocks should be ready to adapt always when changes in markets, customer needs, or strategy occur and then create new, redirect existing, or dry out old value streams. (Knaster & Leffingwell, 2020, pp. 103–106)

3.2.2 Competencies

In addition to lean-agile principals, SAFe presents seven competences which should be harnessed to achieve business agility, which is the ultimate goal of

the transformation journey for the adopting organisation (Knaster & Leffingwell, 2020, p. 110). These seven competencies are:

1. Lean-agile leadership
2. Team and technical agility
3. Agile product delivery
4. Enterprise solution delivery
5. Lean portfolio management
6. Organisational agility
7. Continuous learning culture

Successful agile transformation requires commitment, strong support and lean-agile leadership, the first of SAFe's competences. Lean-agile leaders can create an environment where transformation can begin in agile teams and spread to rest of the organisation by allowing the collective expertise of teams to flourish and make the best decisions. Lean-agile leadership is summarised in three dimensions: mindset and principles, leading by example, and leading change. (Knaster & Leffingwell, 2020, pp. 111–112)

The second competence, team and technical agility instructs what an agile team is and how one should create value in harmony with others like it (Knaster & Leffingwell, 2020, p. 123). Agile teams are accountable and cross-functional, meaning that they harness five to eleven people with authority and the required skillset to produce value in iterations. This is done by utilising the skillset provided by the diverse skill pool of its members. (Knaster & Leffingwell, 2020, p. 125) Agile teams in SAFe utilise familiar elements from Scrum and XP as they contain similar roles and responsibilities as Scrum (product owner, scrum master, team members) and are guided towards development practicalities of XP. A team has their own backlog, which is a collection of user and enabler stories or other development items which are prerequisites to continue value delivery. User stories express desired functionality and so are value-centric whereas enabler stories lay groundwork for efficiency towards future feature development. (Knaster & Leffingwell, 2020, pp. 126–127)

Iterations in agile teams last one to two weeks during which the developed system in continuously integrated, evaluated and tested. An iteration starts with a planning session where to-be worked stories are discussed, polished, split if needed, and projected along with their criteria of acceptance. As in Scrum, the team meets daily to sync up with progress during the iteration. (Knaster & Leffingwell, 2020, pp. 130–131) In the end of an iteration a high-quality system increment is delivered in accordance with the stories which were selected for the iteration and were also demoed. A retrospective session is held to gather opinions how the iteration went and what could be done better next time. This enables continuous improvement for the development process. (Knaster & Leffingwell, 2020, p. 131)

In large organisations, single teams often cannot complete systems on the scale that is required because they require more skills and scope than a single

team can possess. This creates the need for multiple agile teams. One of SAFe's objectives is to coordinate teams of agile teams to produce value in synchronisation along value streams to realise their potential. Agile teams are situated in Agile Release Trains (ARTs) of 50 to 125 people overall. An ART and its agile teams share a common mission and develop and deliver solutions incrementally with the help of practices that guide it to build quality in. Some of these include ensuring continuous flow of work by limiting the amount and size of working items, peer review by conducting work in pairs to gain feedback, allowing anyone to make changes to improve quality through collective ownership, automation of manual tasks, and defining a set point when the item is considered done. (Knaster & Leffingwell, 2020, pp. 135–137)

Value streams and Agile Release Trains (ARTs) are the two most important building blocks SAFe's implementation builds on (Knaster & Leffingwell, 2020, pp. 295–296). The ART is the step above agile team level and has its own roles to keep it going. These include a Release Train Engineer (RTE), Product Management, System Architect/Engineer, Business Owners, and of course, Customers. The ART's role is to "facilitate value flow" with systems thinking and organising around value. The agile teams of an ART all apply the same iteration cycle to their planning, demo and learning events to benefit from simultaneous cooperation between other teams and individual autonomy in finding the best solutions to deliver value and enable iterativity to the system level. (Knaster & Leffingwell, 2020, pp. 133–134)

The most relevant pair concepts behind SAFe's third competence, Agile Product Delivery are Customer Centricity and Design Thinking, Develop on Cadence and Release on Demand, and DevOps and the continuous delivery pipeline. They further tie together how SAFe sees work should be structured in a large environment. Together these aspects characterise, develop and release products of value continuously to customers. (Knaster & Leffingwell, 2020, pp. 141–143)

Businesses should be able to focus their execution by building "desirable, feasible, viable and sustainable" solutions keeping customer always on mind with design thinking and targeting the right customers with the correct timing with developing on cadence and releasing on demand. Design thinking incorporates a concept called a double diamond: in the first diamond the aim is to understand the problem, and in the second to design the right solution based on the first diamond. (Knaster & Leffingwell, 2020, p. 144) The best time to release these solutions to market is not always as soon as they are ready, because some organisations may need time to ensure legal compliance or suitable market conditions overall. Therefore, SAFe explains how separating releases from development rhythm may be beneficial to some organisations. (Knaster & Leffingwell, 2020, pp. 141–142)

Continuing the specifics of the third competence, developing on cadence but releasing on demand intends organisations to offer a non-stop value flow to its customers with solutions, which have been designed for them from the ground up. SAFe targets value delivery of ART(s) to bigger Program Incre-

ments (PIs), which contain their iterations and therefore deliver value incrementally with working and tested results. PIs typically last eight to twelve weeks, during which there are three to five iterations. Durations of PIs are very flexible and can be adjusted to the needs of the organisation and its target market similarly to releases, which can take place anytime as an PI progresses. (Knaster & Leffingwell, 2020, pp. 151–152)

A program in SAFe has its own backlog, which is where features that fulfil user needs, and provide business benefits wait to be picked for a PI to be further assigned to development. Features are not products, but smaller parts of them responding to a desire, which are opened with benefit hypothesis. Features are supposed to be sized to fit to a single PI or be finished during it in a shorter time. Similarly with stories earlier, features also have enabler features which allow building sustainable solutions in a longer run. There are also non-functional requirements (NFRs), such as performance and security, that persist on the backlog to steer development towards always fulfilling them. (Knaster & Leffingwell, 2020, pp. 152–153)

The fourth competence of SAFe, Enterprise Solution Delivery, aims to tie lean-agile ways together with developing and delivering massive but advanced solutions – a combination often deemed incompatible (Knaster & Leffingwell, 2020, pp. 177–178). From the previous team and program levels, it moves one step above to the solution level, which has its own roles, solution trains and backlog. Coordination of these and its other structural concepts is again meant to follow value streams, which on this level may be very complex. (Knaster & Leffingwell, 2020, p. 190)

Ultimately development always comes down to money to acquire resources. The enterprise always has the final say how money is distributed among the development value streams of portfolios (Knaster & Leffingwell, 2020, p. 208). The fifth competence in SAFe, Lean Portfolio Management (LPM), utilises Lean and systems thinking principalities to strategy and investment financing to support optimal strategy and execution alignment. As a term, portfolio management itself focuses on steering development and releases to follow strategy. In SAFe, traditional portfolio management is modified to lean-agility to operate in modern business environments. (Knaster & Leffingwell, 2020, pp. 205–206) LPM has its own set of roles, some of which include Business Owners, Epic Owners, and Lean-Agile Centre of Excellence (LACE) (Knaster & Leffingwell, 2020, p. 209).

There are three dimensions in LPM, the first being strategy and investment funding. It makes sure that development which supports the company's strategy receives funding, therefore aligning the portfolio correctly. It highlights co-operation between Business Owners, Enterprise Architects and other portfolio stakeholders. (Knaster & Leffingwell, 2020, pp. 209–210) The most important portfolio investments are called Epics, which span across value streams and may take several PIs to finish. Their conceptualisation takes inspiration from Lean start-up in true Minimal Viable Product (MVP) spirit to first test hypothe-

sis and through evaluation determine next best actions for the epic. (Knaster & Leffingwell, 2020, pp. 118–121)

Agile portfolio operation in LPM has three responsibilities: "Coordinate value streams, support program execution, and foster operative excellence." Successful value-stream co-operation can bring intense competitive advantages for the company. Supporting program execution gathers the best practices around the portfolio for everyone to harness. Finally, fostering operative excellence boils down to continuous improvement of operations. Its main driver is the LACE. (Knaster & Leffingwell, 2020, pp. 226–228)

Lean governance in LPM is listed to oversee and estimate spending, ensuring "audit and compliance" and measuring the current operations. In budgeting it accounts for uncertainty by allocating costs with story points in smaller segments. Ensuring continuous compliance observes that obligatory standards and regulations are enforced not only in the end of development but during it. Measuring portfolio performance incorporates relevant KPIs (Key Performance Indicator) of value streams to measure their status to estimates. (Knaster & Leffingwell, 2020, pp. 229–233)

The sixth SAFe competence is organisational agility. It once again promotes the benefits lean-thinking and agile teams bring for the company in today's fierce markets where adaptation to survive is gained with organisation agility. Lean agility must empower the whole company which starts from people believing in its fundamentals. Creating great solutions is not enough, and companies should satisfy the "customer journey" from start to finish. (Knaster & Leffingwell, 2020, p. 259)

Continuous learning culture is the last, seventh principle of SAFe. Learning empowers individuals of a company to "increase knowledge, competence, performance, and innovation" to stay ahead of the competition. This is done with dimensions of learning organisation, innovation culture, and relentless improvement. Learning continuously attaches these benefits to uncertain, and the fast nature of today's world. (Knaster & Leffingwell, 2020, pp. 260–262)

Values and principles of Agile manifesto, Scrum, XP, SAFe 5 and Lean software development are laid out on Table 1. Many similarities can be spotted with each method adapting similar fundamentals to their own takes on agility. Lean software development does not directly list its values, but it is founded in of lean thinking (Poppendieck & Poppendieck, 2003, p. 9), values of which are included in the table. Also, Scrum does not include a list of unambiguous principles in its reference material, hence its principles section is blank in the table.

Table 1 Comparison of values and principles of Scrum, XP, Lean software development and SAFe 5 (Beck, 1999a, p. 35; Beck et al., 2001; Knaster & Leffingwell, 2020, pp. 72 and 116; Poppendieck & Poppendieck, 2003; Schwaber & Beedle, 2002)

	Values	**Principles**
Agile manifesto	Individuals and interactions over processes and tools Working software over comprehensive documentation Customer collaboration over contract negotiation Responding to change over following a plan	Satisfy the customer with continuous delivery of valuable software Welcome change and harness it Deliver working software frequently Tight cooperation between business and developers Motivated people with trust and support get the job done Face-to-face communication is the best
Scrum	Commitment Focus Openness Respect Courage	
XP	Communication Feedback Simplicity Courage Respect	Rapid feedback Assume simplicity Incremental change Embracing change Quality work
SAFe 5	Alignment Built-in quality Transparency Program execution	Take an economic view Apply systems thinking Assume variability; preserve options Build incrementally with fast, integrated learning cycles Base milestones on objective evaluation of working systems Visualise and limit Work in Process (WIP), reduce batch sizes and manage queue lengths Apply cadence; synchronise with cross-domain planning Unlock motivation of knowledge workers. Decentralise decision-making Organise around value
Lean software development	Value Value stream Flow Pull Perfection	Eliminate waste Amplify learning Decide as late as possible Deliver as late as possible Empower the team Build integrity in See the whole

3.3 Challenges of scaling agile

Implementing a scaled agile method is a radical process and difficult. There are many moving pieces and challenges when fundamental operations are transformed to be based on agile principles (Kalenda et al., 2018). For example, a common comprising challenge for agility-adopting companies is "doing agile" instead of actually "being agile" (Ranganath, 2011). This expression sums the overall challenge of agility very well. Practices such as purchasing tools that claim to enable agility to organisation are easy for management, but do not transform structures and ways of working towards true agility where the real heavy lifting lies.

Scaling agility successfully is a delicate balance of authority between top-down planning and autonomy of truly agile teams (Conboy & Carroll, 2019; Moe et al., 2021). Traditionally, heavy top-down planning activities have been used to tame complexity of large projects. While this may sound practical, projects tend to change and evolve when they are active and become even larger and more complex (Klakegg et al., 2016). Strict plans do not address emerging changes well, which is why creating solutions agilely has been viewed as the remedy to surprising alterations. In a large-scale environment coordinating development of multiple teams towards a unified goal has typically been achieved with "organisational control" which conflicts with team autonomy, fundamental principle of agility. (Moe et al., 2021)

Large-scale projects require more people and stakeholders to be involved, which increases interconnectivity and makes team autonomy more difficult to achieve when there are stakeholders outside the team whose input is needed (Šmite et al., 2017). Even though one principle of agile teams is to bring all the needed stakeholders of development close to one another to enable different experts to pitch in always when needed in the development cycle (Beedle et al., 1998), in a large-scale environment, this may be impossible to achieve (Šmite et al., 2017). Moe et al., (2021) describe that autonomy of agile teams in large-scale development "cannot be limitless".

As a logical consequence to all this, scaling agile is seen difficult to achieve in globally distributed development where teams are separated by long geographical distances and communicate via electronic means (Paasivaara, 2017). Extending the effect even more, Figalist et al., (2019) found in their conference paper that the same challenges also apply to scaling agility outside organisational borders. This was mainly due to "long communication paths" or lack thereof altogether. They also found how these long distances could even provoke competition between teams, which affected their co-operation negatively (Figalist et al., 2019). Long communication can be considered particularly prone to the effects familiar from the children's game *telephone*, where true intentions of message disappear through many hops in the chain of communication.

Conboy and Carroll (2019) present pitfalls that they recognised organisations adapting scaled agile methods may fall in. They describe how lack of sub-

stance in scaled agile methods creates uncertainty for organisations what to specifically do, which can lead to misinterpretations and/or resorting to old familiar, non-agile ways. This is very understandable, because we as humans do not like uncertainty and it is always easy to continue doing something how it has always been done.

Conboy and Carroll (2019) also found that choosing the right framework for the organisation was difficult due to the lack of comparisons between methods. Trying out different frameworks and their practices may also lead to fatigue and frustration in employees and hence is discouraged. They point out how research of scaled agile methodology lacks real-world example studies instead focusing on specific contexts where they severe were issues. Organisations may also get stuck on measuring correspondence of their operations to the selected framework instead of the produced value, which is the real indicator of success or failure. The optimal degree of transformation should also be discussed, if for example, transforming the last remaining pieces proves to be exponentially expensive or in the best case not even mandatory for other operations. In some cases, the customer orientation and involvement in the development process also made it impossible to work according to the scaled framework when the mismatch between parties was too great to converge. These individual alterations, despite being temporary, to or from the transformed development model proved understandably extremely undesirable. (Conboy & Carroll, 2019)

Van Wessel et al., (2021) describe how Enterprise Architecture (EA) and scaled agile methods in their default configurations are rather incompatible with each other. Tamm et al., (2011) define enterprise architecture as "the definition and representation of a high-level view of an enterprise's business processes and IT systems, their interrelationships, and the extent to which these processes and systems are shared by different parts of the enterprise". Van Wessel et al., (2021) elaborate that if an organisation has already established EA functions, fitting them together with adoption of a scaled agile framework is challenging as they do not provide much, if any guidance on how to do so successfully. Considering this to the earlier definition of EA, it is only natural that these two concepts may present conflicts to one another, particularly if their coexistence and -operation are not accounted for. Therefore the authors underline the importance of sufficient preparation when fitting the two together in an organisation to avoid conflicts. (van Wessel et al., 2021) In a similar fashion, Horlach et al., (2018) argue in their conference paper that IT governance efforts towards optimal business-IT alignment in companies and agile frameworks do not find common ground. This is mainly due IT governance being largely reliant to long-term planning while agility unleashes its benefits with team-level autonomy (Horlach et al., 2018).

Scaling agile in specific domains and contexts has proven to be more challenging than it normally is. One of the instances where methods face more challenges is development of safety-critical systems in regulation-heavy markets (Fitzgerald et al., 2013). Another observed characteristic of a particularly chal-

lenging environment for agility is a heavy bureaucracy (Hobbs & Petit, 2017). This is only understandable given the contradictions and therefore difficulty of proper alignment between self-directing autonomy and planning in large organisations. Continuous deployment and delivery of value have also seen challenging in a scaled agile environment, because they depend on a unified understanding of what customer value really is (Kasauli et al., 2017).

Moreover, in their conference paper Eklund et al., (2014) found how adopting scaled agility poses additional challenges in development of embedded systems. First, they propose that fast pace of iterations in agile development is problematic to the longer and varying timeframes it takes to produce a hardware product with varying embedded (software) features from start to finish. They also present similar remarks as Fitzgerald et al., (2013) that agile methods do not incorporate strict security requirements fundamentally to them, which embedded products typically necessitate. Their business model which is reliant on a hardware platform increases interconnectivity and also makes customer involvement more difficult requiring more forward planning (Eklund et al., 2014).

In their two-fold research, a literature review and a case study of implementation of a scaled agile method, Kalenda et al., (2018) identify several major challenges in the way of a success. In both parts challenges regarding change resistance, quality assurance, and conflicts between non-agile and agile units were observed. Resistance to change was found to root from people who disliked the new practices despite of being familiar with agile development overall. Development teams which obtained full autonomy did not want the change their existing processes and tools. This touches on the limitless autonomy issue by Moe et al., (2021) discussed earlier. Perhaps a gentle push is needed to kickstart agile adoption? Conflicts emerged when agile units of the organisation were merged with non-agile ones and implementation of the method was also experienced to have progressed too fast. Restructuring the organisation to support agility also posed severe challenges when moving resources from a place to another created misfits in their supply and demand. Additionally, quality assurance issues emerged when total autonomy of teams gave birth to new processes which were had not yet matured. (Kalenda et al., 2018)

While not directly a challenge, an interesting phenomenon concerning agility and its scaling is its general commercialisation today. The agile manifesto by its tone was a joint statement from influential and passionate industry pioneers pushing for a better way of working in the software industry. Since then, their ideals have been bundled in various methods, and most recently in scaled methods, such as SAFe. The nature of, for example, SAFe has attracted a lot of criticism partly due to its "consultant-like" takes and commercial nature (*The SAFe Delusion – Information for Decision-Makers Considering the SAFe Framework*, 2022). Overall, it has been claimed that the true essence and "diversity" of agile has been forgotten in place of blind trust in a few methods, which are not understood in the depth they would require (Hohl et al., 2018; Klünder et al., 2017).

On the other hand, it is always easy for a framework to put the blame of a failure on a lacklustre implementation effort of the organisation. Commitment is a word often used, which of course is required. But who is to blame of failure, incompatible framework, domain, insufficient implementation/commitment, or something else? Frameworks have shiny promises and adopting one may sound as easy to top management as purchasing a new fancy tool, where it is a simple act of using it to gain its benefits. In the case of agile transformation, this however is not nearly enough. The adopting organisation must be ready to view itself critically and be brave to investigate itself to recognise working concepts and demolish old ways that do not complement agility – or do the opposite to customise the selected method to their operations.

A summary of the challenges discussed in this section is presented in Table 2. Literature is grouped to four categories based on the challenge(s) they discuss. Methodical challenges deal with issues of scaled agile methods themselves and shortcomings in the specification or implementation. Cultural challenges contain transformation, alignment, and other types of conflicts which organisations may encounter when transforming to agility from established operations. Domain-specific challenges are issues that are isolated to only certain environments. Challenges regarding complexity root from the vast interconnectivity and number of stakeholders present in large organisations.

Table 2 Grouping of scaled agile literature based on their type of recognised challenges

Types of challenges	**References**
Methodical challenges	Kasauli et al., (2017) Conboy & Carroll (2019) Kalenda et al., (2018)
Conflicts with other methods	Van Wessel et al., (2021) Horlach et al., (2018)
Cultural challenges	Ranganath, (2011) Kalenda et al., (2018) Moe et al., (2021) Hobbs & Petit, (2017) Edison et al., (2022)
Domain-specific challenges	Fitzgerald et al., (2013) Eklund et al., (2014) Hobbs & Petit, (2017)
Complexity challenges	Šmite et al., (2017) Moe et al., (2021) Figalist et al., (2019)

4 CUSTOMER VALUE IN AGILE

This chapter is dedicated to investigating and explaining what customer value is to increase understanding of it and what factors contribute to its accumulation. Another angle is to research how a concept as abstract as customer value can be measured. Also, other metrics utilised in measuring agile development work itself and customer experience, a sister term of sort for customer value, are covered. Research of these topics is scarce and very scattered with comprehensive studies remaining practically non-existent. Great deal of studies also appears as conference papers, which is telling of the immaturity of the subjects despite their profound and concurrent status in practice and historically in agility itself. Some agile methods have built-in measurements and backlog prioritisation mechanisms or take inspiration of more common methods, which are described with their relative objectivity.

4.1 Components of customer value

Customer value is a hot topic in today's companies. It is very common for organisations to promote and call themselves customer-centric or -oriented while pursuing agility to increase competitiveness (Conboy & Carroll, 2019; Ebert & Paasivaara, 2017). Setting aside customer value's almost buzzwordy connotation allows studying what it really means together with its integral ties to agility. While there are a plethora of definitions for customer value, one of the most cited ones in its literature is presented by Woodruff (1997):

> *Customer value is a customer's perceived preference for and evaluation of those product attributes, attribute performances, and consequences arising from use that facilitate (or block) achieving the customer's goals and purposes in use situations.*

This definition covers the formulation of perceived value extensively, although having a particularly rational logic to customer's behaviour by using words of

"customer's goals and purposes". Perhaps these two can be considered to include more irrational behaviour as well. The topic of customer value has been researched for over three decades particularly in marketing domain (Gallarza et al., 2011), where three distinct schools of thought have been recognised: positivist, interpretive, and social constructionist. Zeithaml et al., (2020) categorise differences to five common dimensions where they seek answers:

1. Who and what the sources of value are? Individualistic vs. contextual perception.
2. How can value be researched methodologically and "captured"? Value through perception, experience, or co-creation.
3. How do "value dimensions" interact with each other, different conditions, contexts, and new technology?
4. When and where does customer value appear in complex systems and interactions?
5. To what extent does customer value obey contingencies from its environment, such as business performance, customers' well-being, or their emotions?

As we can notice, customer value is a multifaceted topic in academia, where its research has sprung into several lines of thought addressing relevant questions from their unique viewpoints. In agile development, value as a term has seen some research and is generally attributed to business value, with common perception remaining vague. Perceptions of the meaning and most important aspects of value in agile organisations differ across domains, where for example in the telecommunications domain, perceived quality has been identified as a prioritised component in value creation. (Alahyari et al., 2017) This can be seen natural for incumbent companies in domains with established products and knowledge of their customers. There are surely as many different perceptions as there are companies with unique customers and their value perceptions. Then again, companies only have a mere grasp of what their customers desire, which still can be considered as a lot more precise than the generalisable scope of research papers.

A sister term close to customer value, customer experience has its own stream of research and is often discussed in the broader context of customer value. The paradigms of customer value commonly recognise customer experience serve as a subcomponent in overall customer value with their own viewpoints. (Zeithaml et al., 2020) Similarly to customer value, it does not have a unanimous definition with different branches of research attributing different things to it or handling it from altering viewpoints. Becker & Jaakkola (2020) identify three branches to its definition, with the first viewing it in parallel with resulting effects, such as satisfaction or value. Some see it as a sole factor which provides satisfaction independently, with the third school of thought attributing customer experience closely to a product as its trait. Based on these distinctions the authors propose a new definition for customer value as "*non-deliberate, spontaneous responses and reactions to particular stimuli*". (Becker &

Jaakkola, 2020) These characteristics make it possible to assess customer experience as a subcomponent of customer value and therefore to extract interesting and relevant findings from its research.

Another critical point to clarify is who customers really are. Are the end users of the product customers, or is the development internal development in the host organisation, or perhaps a combination of various customer-like stakeholders? In B2B environments, customers have typically been seen as "purchasing agents", the "customer company", or users of this customer company (Peppers & Rogers, 2017, p. 21). Creating the best value for customers in B2B environments has been found to require "resource integration" between the contracting firm and its service provider(s) (Macdonald et al., 2016). The point of organisation-specific perception of value can also be applied to this question, as the nature of customers is dependent on the company. Development for internal and external customers does not exclude one another, a combination of them on varying degrees surely being the most prevalent in organisations.

The increased competition which digitalisation has accelerated pushes companies to increase their cost-effectiveness and diminishing returns from product customisations do the opposite. This means the companies aiming to measure and increase customer value must align value through customisation with internal efficiency which customised offerings typically hinder (Squire et al., 2004). These trends have hit telecommunications providers, among which providing the best possible customer experience has become an increasingly vital competitive differentiator. As growth through innovation has slowed down and prices have entered a downwards trend due to increased competition, the pressure to organise operations efficiently has increased (Penkert et al., 2019). In a market where service providers cannot differentiate themselves with their offerings, other means such as customer experience become increasingly crucial to attract business growth.

A logical method to increase customer value of development is to gather requirements and feedback directly from them. While beneficial, this however poses the risk of stretching the scope of the project as customers may wish for features that would be infeasible or even unrealistic for the supplier to implement (Ashmore & Wedlake, 2016). Afflerbach and Frank (2016) explain how customers are no longer bound to few suppliers in markets and instead, suppliers are increasingly at the mercy of customers to attract business. Customers may wish features that require extensive customisation of products, which can quickly inflate the project's costs substantially, and particularly in the case of software, decrease their natural scaling potential and therefore the bottom line.

Despite Woodruff's (1997) popular definition, a more recent study by Gallarza et al., (2011) concluded that there is no established, unanimous conceptualisation of what customer value really is. As the result, standardised means of measurement have not been adopted. Customers can view value in several different aspects, which the same authors have gathered from customer value research. They are: Perceived quality, perceived price, perceived value, satisfaction, and loyalty. (Gallarza et al., 2011) Each of these can be thought to make

assigning numerical values to the term customer value even more challenging. This is because awareness of these differences in value perception decreases the measurement's reliability and comparability between detailed environments.

4.2 Measuring customer value

Measuring objects is something we do all the time in our daily lives. It means attributing comprehensible numbers, or even symbols, to objects following a certain set of rules to reflect them in a concise and information-rich form. This increases the way they can be further explained and therefore understood in detail (Fenton & Bieman, 2014, p. 27; Finkelstein, 1982). Measuring can be seen as the basis of improving the existing, because it offers means to compare effects of actions to the target of measurement (Fenton & Bieman, 2014, pp. 27–28). However, measuring a concept as abstract as value is fundamentally difficult because it does not have easily measurable attributes but is instead based on more sentimental aspects of us humans. Therefore, assigning value from and to individual items of their value is not straightforward. We can view customer experience to shed light on the difficulty of customer value's measurement due its proximity to customer value and thanks to it having existing and proven metrics.

Customers are one, if not the most important aspect in the agile paradigm. This is apparent from customers being one of the first points of interest in the core statement of the agile manifesto (Beck et al., 2001). Customer value being as broad of a term as it is, questions relating to the specifics of the environment where it should be increased or measured come to mind effortlessly. In the context of agile development targeting increased customer value, is the developed item responding to customer feedback, demands and/or wishes? Is it based on a perceived increase on customer satisfaction when the development is ready? These question can be seen to fit to the earlier customer value definition by Woodruff (1997) But what if the development item related to legal/regulatory compliance which does not provide any customer value by itself, but if neglected, can render the product illicit and prevent customer value creation altogether? Or if it is a security feature that upon negligence can destroy customer value if security issues infect the product? These questions further highlight the complicated nature measuring customer value.

Measuring the impactful subcomponent of customer value, customer experience is a prominent topic of its own both in academia, and company contexts, where it has seen plentiful research and use (Becker & Jaakkola, 2020). Perhaps the most influential metric of customer experience is the Net Promoter Score (NPS), which was invented in 2003 by Fred Reichheld at Bain & Company. It comprises of a simple question presented to customers: how likely are you to recommend the organisation to a friend?" There are three categories where customer responses to this question are categorised to:

1. Promoters (scores of 9 and 10)
2. Passives (scores of 7 and 8)
3. Detractors (scores between 0 and 6)

The total NPS score is calculated by subtracting detractors from promoters. (Reichheld & Markey, 2011) Reichheld (2003) has gone as far as to claim that NPS defeats all other in significance by attributing the likelihood of recommending the offerings of an organisation as an all-encompassing metric of customer satisfaction and even revenue.

4.3 Value through prioritisation

Realising the buzz of customer centricity is a crucial topic for many organisations and is what the target company of this thesis as well is aspiring trying to improve. As the number of needs and requirements often greatly exceeds the number of resources available to deliver them, prioritisation must be practiced. This has direct implications on the end result, and creates the question how development prioritisation can be based in a on value while not neglecting the bottom line and other influencing factors (Cleland-Huang, 2015). Moreira (2017, p. 139) also describes a common challenge for companies to be that their development funnel becomes congested due to development items being becoming too large or having too many items on the table at once. This sets a new challenge for organisations to solve: how could development items be prioritised based on their customer value in reasonable portions?

In technical development, such as coding, there are many more attributable numbers which can be utilised in measurement. These include numbers from amount of bug reports, for example (Kupiainen et al., 2015). Conference paper by Storti & Clear (2020) found the most used product measures in agile software development to be sprint velocity, story points, and number of defects. The same authors point out that popular measures focus on measuring the agile development process itself, rather than the product. This presents an obvious contradiction with the spirit of agility, where improving the development process certainly is an important aspect but targets a better outcome regarding the product in the bigger picture, rather than sole process efficiency. One can also consider that that focusing on the sole amount of development output instead of actual outcome may also mislead the effectiveness of development and perception of value.

Process development activities have more previously explained "sentimental components" that are notoriously difficult measure in numerical values. Regarding customers or their value, customer satisfaction was among the least popular measures (Kreuzer et al., 2020). Additionally in the context of scaled agile methods, the same researchers were not able to find an impactful number of papers to study the topic. Sambinelli & Borges (2019) found how proven metrics for measuring customer value are almost non-existent in research.

It is commonplace for conditions to change and certain development items to become more urgent than others (Knaster & Leffingwell, 2020, pp. 152–153). Companies must find the most (customer) value-rich development items to put their scarce development resources to the best use, as rarely are these resources abundant (Lehnert et al., 2016). Generally in agile methods, such as Scrum, it is seen as part of the product owner's role to order the product backlog from the most value-rich items first to the least (Schwaber & Beedle, 2002). In scaled agile development however, the product owner may not be able to objectively handle the viewpoints of all development stakeholders and so fail to maintain optimal prioritisation, according to a conference paper by Olsson & Bosch (2015). In their earlier conference paper Olsson & Bosch (2014) call for the need of re-prioritisation activities to occur continuously during development to avoid a once greenlit decision becoming the sole driver of development instead of evaluating its customer value. Power (2011) calls the limited viewpoints in organisational silos another potential pitfall of value delivery as they are focused on their own efficiency instead of value delivery to customers. He adds how solely staring at numbers of only the next business cycle is detrimental for value improvements which require investment in a bigger picture of development.

Alongside the vision of the product owner or other development stakeholders, qualitative measures to back prioritisation decisions exist (Moreira, 2017, p. 138). Prioritisation must always be done to ensure continuity in development flow without too many items on the table at once to maintain continuous delivery of customer value (Moreira, 2017, p. 139). Cleland-Huang (2015) describes many existing methods, such as assigning points or classifying priority ranks (e.g. high/low) to development items prone to biases of the people who are creating these assignments in those situations. This underlines the importance for methods which incorporate diverse metrics and so can be used as comprehensive backing tools for prioritisation decisions.

SAFe instructs product management to be responsible for the order of the backlog items and proposes Weighted Shortest Job First (WSJF) model to incorporate product development economics to prioritisation tasks. In SAFe's instructions for using WSJF, Cost of Delay (CoD) of an item is the ruling metric where it is divided by its relative size. The sum of User-Business Value, Time Criticality, Risk Reduction & Opportunity Enablement make up CoD, where one's smallest assigned value is one and is used as a point of reference to other values in relation. When CoD is calculated, it divided with relative job size to get the WSJF score of a backlog item. The higher the resulting value is, the higher the item is on the backlog. WSJF generally favours small but value-rich items. (Knaster & Leffingwell, 2020, pp. 154–155)

Cleland-Huang (2015) set prioritising parts of development that can empower learning and delivery of real customer value as the main goals of making prioritisation decisions with strong and objective backing. It is strongly inspired by Minimum Viable Product (MVP) thinking by Ries (2011), where its intention is to see the first minimal version as a base for feedback to guide development decisions towards the best customer outcome. It begins with creation of user

stories, which are mapped and then set to a project backbone based on their criticality. Next, the stories are prioritised to the "big-bang approach" to deliver essential features based on positive or negative revenue of the feature. For example, revenue is negative when the feature is developed, but positive when the feature can provide revenue once it is shipped, such as billing the customers. This builds the solution's barebones MVP version that has the necessary components to get it up and running, therefore including components which do not directly generate revenue but are still mandatory. The next phase focuses on delivering functionality which increases the revenue stream to the point full functionality has been delivered. (Cleland-Huang, 2015)

Hannay et al., (2017) aim to solve the missing link between spending and development impact by presenting benefit points. They are assigned based on the perceived business value the corresponding project (epic) will bring when it is ready and deployed. Benefit points work together with story points which indicate costs of the project. They complement the project's business case (the goals of the project with go-approval) which links it to strategy and vision. Benefit points reflect the role of an individual epic relatively towards the goals of the project in the bigger picture. The authors call benefit points a considerably simper tool than SAFe's WSJF, which relies in multiple metrics which can be difficult to valuate. (Hannay et al., 2017)

5 SUMMARY AND DISCUSSION OF LITERATURE REVIEW

This closing chapter of the literature review is dedicated to summarising main points of the previous chapters to answer research questions 1.1 and 2.1 of this thesis. These questions were planned to be answered according to research literature of scaled agile challenges, customer value and its role in development activities. This goal was reached but not without considerations surfacing during the research process, which are also explained to increase the rigor of discussed results. The summarisation and discussion serve to provide a concise background for the following empiric study.

Based on the research referenced in this literature review, both agile and scaled agile concepts and methods alongside their challenges have their unique and established streams of research. The fundamentals of agile software development lie in the agile manifesto from 2001, the initial concepts of which have inspired practitioners even long before (Beck et al., 2001; Larman & Basili, 2003). Since the first appearances of agile principals and the manifesto, the software industry has slowly but surely shifted towards agility, which nowadays is the de facto standard instead of previous plan-driven models (Rodríguez et al., 2019). Agile methods have even inspired other domains to adopt their principles outside the software industry, which is telling of their success (Hohl et al., 2018). Agile software development has also itself taken inspiration from elsewhere, such as Lean thinking from manufacturing industry (Ebert et al., 2012).

Key concepts behind agility's benefits centre around iterativity and incrementality, which push to build solutions piece by piece while learning from previous cycles and to accept changes to the big picture (Larman & Basili, 2003). Combined with changes in the management style to empower and harness the expertise of boots on the ground inspired agility's takes on software development (Beedle et al., 1998). These fresh principles were brought to practice with new agile software development methods such as Scrum, Extreme Programming, and Lean software development (Rodríguez et al., 2019). Today, Scrum is still the most popular team-level agile method in the industry (*16th State of Agile Report*, 2022).

The original methods such as Scrum and their fundamentals have typically been perceived as fitting to small organisations and newcomers, such as start-ups that harness agile benefits by their organic form. In the same vein, their small ideal team size makes them difficult and infeasible to adapt to larger endeavours. (Ebert & Paasivaara, 2017; Kalenda et al., 2018; Rigby et al., 2018) As a solution, scaled agile methods have been created which aim to deliver the benefits of agility to large organisations in their efforts to become agile players to maintain and increase competitiveness (Ebert & Paasivaara, 2017).

The most used scaled agile method is the Scaled Agile Framework (SAFe) (*16th State of Agile Report*, 2022), which presents its contents through seven competencies. When they are successfully harnessed, SAFe promises companies to increase their readiness to change in their operating environments, deliver value to customers incrementally and resourcefully while also making sure their products and solutions live up to quality standards. This agile transformation can be viewed as the path towards true, digital transformation with means of agility. (Knaster & Leffingwell, 2020, pp. 23–24) The latter's high-level definition by Fitzgerald et al., (2014), utilising technology to enhance existing business, certainly becomes more reachable if agile transformation's goal of preparing an organisation for modern business environments with learning and adaptability (Knaster & Leffingwell, 2020, pp. 23–24) is reached. In any case, an agile transformation requires serious commitment and courage from the company to conduct successfully (Knaster & Leffingwell, 2020, pp. 111–112).

Despite the glorious claims SAFe presents of its benefits, many challenges have been identified around scaling agility and its methods in general (Conboy & Carroll, 2019; Kalenda et al., 2018; Paasivaara et al., 2018). Somewhat ironically SAFe itself has been criticised as "too heavy and complex" by industry professionals (Ebert & Paasivaara, 2017; *The SAFe Delusion – Information for Decision-Makers Considering the SAFe Framework*, 2022). In some parts, the language of SAFe's manual resembles the tone of a sales pitch. This in turn provokes curiosity of the framework's true efficacy in action. One would presume that the flashy pitches would be left to the homepage, and instructions would focus on more hands-on descriptions. Then again, perhaps the radical nature of agile transformation requires even its guidebook to ensure its reader of its upsides continuously. In practical environment, the commercialisation of agile methods has also raised scepticism, particularly around SAFe, with the lack of awareness of the diversity of agile method a present concern in the industry (Hohl et al., 2018; *The SAFe Delusion – Information for Decision-Makers Considering the SAFe Framework*, 2022).

However, despite heavy critique towards SAFe, all blame cannot be put on it. Scaling agile is almost notoriously difficult as there are numerous challenges where embarking on the agile transformation journey can fail (Rigby et al., 2018). On a fundamental level, these challenges often root from the optimal alignment methods seek between team-level autonomy, which provides agility's benefits through better low-level decisions, and forward-planning activities (Conboy & Carroll, 2019; Moe et al., 2021). Planning is used to manage the com-

plexity companies face from their size and interconnectivity (Klakegg et al., 2016).

The details and practicalities of adopting a scaled agile framework and above all, agile transformation, is the organisation's responsibility to assess optimally with the help and inspiration of the selected method (Ebert et al., 2012). As complex as modern companies are it really is no wonder how transforming them from established ways is as challenging as the research of the issues suggests. Agile methods should not be considered magical, one size fits all remedies. Conboy and Carroll (2019) go as far as to question whether scaling methods can even be called methods due to their lack of substance. To utilise them to their full potential, they and their implementation environments must be known thoroughly instead of trusting new tools to take care of the change (Ebert & Paasivaara, 2017). Above all, to become truly agile, a cultural transformation is necessary (Ranganath, 2011). Not easy for incumbents, whose status quo can consequently pave way for their demise, or on a lighter note, weakened market position. The types of observed challenges of scaling agile from its literature were categorised in Table 2 based on their characteristics, providing concise information to answer research question 1.1.

In their natural ambition, companies are trying to provide the maximum value to their customers. This creates the need for them to allocate their finite development resources effectively to produce the maximum amount of customer value (Lehnert et al., 2016). This calls for prioritisation of development items to be based on the value they would provide which typically been assigned for the product owner to handle in agile teams. However, in large environments this may become too vast for one person to manage. (Storti & Clear, 2020)

Measuring value of different types of development, such as process or technical development share few similarities due to the difference in assignable and representable metrics between them (Kupiainen et al., 2015). On the technical side there are many observable and measurable metrics but regarding more sentimental concepts, such as value, such measures do not exist (Kreuzer et al., 2020). Customer value itself has seen significant research interest, where three schools of thought have emerged which discuss the term and its components from varying viewpoints and assumptions. The term itself also has no unified meaning, with common characteristics still encompassing its general conceptualisation in practice. (Macdonald et al., 2016; Zeithaml et al., 2020)

The missing uniformity of customer value research may explain why practices to measure it as the outcome of agile software development are very scarce. The scarcity of proven methods has also been identified in literature by Sambinelli & Borges (2019). Customer experience, which can be viewed as customer value's subcomponent has its own measures. Of these, NPS has attracted a lot of research and is also widely used in quantifying customer experience to representable numerical values (Valentina et al., 2018). Customer experience and its measurement can be used as an assisting link thanks to its relation to customer value, which suffers from a severe lack of proven metrics. NPS has even been viewed as a direct influencer of business revenue (Reichheld & Markey,

2011). Similar remarks of effects on revenue have also been presented of customer value (Cleland-Huang, 2015), which tells that these concepts are close companions in the value delivery of organisations.

Practices towards optimal prioritisation on a more general level have been suggested and are used. Still, it seems that many practices have been targeted for team-level methods with scaled agile methods, such as SAFe, proposing their own practices (Hannay et al., 2017). This is probably be due to the difference in age between these two in research, with interest scaled agile research having soared in popularity only during recent years (Uludağ et al., 2021). The scarceness of proven measurement methods fulfils the research question 2.2, with the difficulty and overall nature of measuring value providing vital backing to the lack of direct research regarding them. It seems that many prioritisation methods take the viewpoint of creating a new solution from scratch, instead of focusing on being implemented to an existing development environment. This hinders the applicability of certain methods as they are and so require customisation.

The scarcity of research regarding customer value measurement in the context of (scaled) agile development models presents a simultaneously challenging, but potential opportunity for the empirical part of this study. This fewness allows investigating the status of customer value measurement and prioritisation from a black box perspective where its best practices are not known. This puts more emphasis on the practical suggestions to spur from observations of the current model alongside characteristics of customer value and experience from their research. Still, the almost chronological order where topics for the subject of this theses have so far been covered in this literature part form a concise picture of the essential concepts and issues around the subject. Countless decisions had to be made to maintain scope withing the range of a master's thesis, which is a natural part of the process, but still forms a limitation to the results as details may always be left uncovered.

6 EMPIRICAL METHODOLOGY

This chapter explains the characteristics of the methodology utilised in the upcoming empirical part. Rationale behind the selection of the chosen method and its characteristics are described through its known strengths and weaknesses to provide a transparent and reliable description of the empirical process overall. This thesis is qualitative by its nature and the first subsection describes qualitative research in general. The empirical data is gathered via semi structured interviews which is explained next. The process of arranging the interviews and creating their contents is also detailed, with thematic analysis being used in the empirical analysis and described in the last subsection.

6.1 Qualitative semi-structured interviews

Oxford English Dictionary defines the word qualitative to refer to evaluation of excellence of a subject, rather than how much of it exists in quantitative terms (OED Online, 2022). This marks the nature of qualitative research, the intention of which is to describe the real world as comprehensively as possible (Hirsjärvi et al., 2009, p. 161). Similar remarks have been made by Robson & McCartan (2016, p. 18), who highlight the difference of social research and particularly, human interactions as subjects of research in contrast to natural phenomena, where fundamental differences in their nature call for methods of their research to differ as well. Qualitative research seeks to "find or discover facts rather than confirm existing claims" of subjects that are difficult or impossible to measure quantitively and therefore require qualitative methods to study them accurately. (Hirsjärvi et al., 2009, p. 161)

Information systems research often deals with both complicated and technical matters together, which hinders effective utilisation of quantitative methods in its contexts and promotes the usability of qualitative ones (Goldkuhl, 2012). Myers (1997) also describes qualitative research to be well fitted to information systems research due to its interest in "managerial and organisational"

topics instead of pure technology. These remarks suit the selection of a qualitative method for this study as its topic discusses both these areas respectively in terms of methods, which are used to arrange development of technological solutions as effectively and efficiently as possible which has a profound human component due to organising how work is structured and performed. Additionally, the nature of the research questions heavily leans on revealing and exploring the current status, rather than approaching the subject with existing claims or hypothesis seeking verification.

A common data collection method in qualitative studies are interviews, which are used to seek answers to questions by asking them from participating interviewees. Different styles of interviews can be roughly categorised to three distinct types: structured interviews which resemble predefined questionnaire surveys, half-structured interviews, and unstructured interviews, that have minimal steering by the interviewer and so allow extensively freeform responses. (Robson & McCartan, 2016, p. 284) While there are many descriptions for a semi-structured interview, in its simplest form, it is a mixture strictly between strictly predefined interview questions and a freeform unstructured interview (Hirsjärvi & Hurme, 2015, p. 47). In practice this means that the interviewer may choose to make relevant alterations to questions or their wording in the interview while preserving their intention to maintain comparability between interviewees. Robson & McCartan (2016, p. 285) describe questions to serve a guiding role in semi-structured interviews, with wording and order of questions being adaptable on a per interview basis by the interviewer's discretion. This includes the possibility for producing follow-up questions on the spot. Semi-structured interviews are the most practiced interview type in information systems research (Myers & Newman, 2007).

Despite and somewhat due to its flexibility, semi-structured interview style also has inherent downsides, some of which are common for quantitative research in general. The simulated environment of an interview may distract its subject, and timely pressure or untrust towards the interviewer may lead to the true nature of topics staying hidden or not reflecting reality. Attracting interviewees from all organisational levels may also be difficult if the interviewer's position is substantially different or conflicts between candidates affect their willingness to participate. These challenges, if realised, can easily lead to biases in the collected data. The interviewer should also grasp the story forming in their head of the data they are acquiring, which may not be as neutral they would presume. True meaning of the interviewee's words may also stay hidden behind specific wordings and the like. Interviews naturally also have the possibility to fail due to unsuccessful use of words by the interviewer or other reasons. (Myers & Newman, 2007)

6.2 Interview planning

The risks of the semi-structured interview method were acknowledged and kept in mind in the drafting of the interview questions and candidate selection. The questions are rooted in the research questions which aim towards empirical findings and thought through the observations of the literature review which were seen to affect the target organisation. This approach by following topics identified from literature is central in semi-structured interviews to support loosely worded questions (Hirsjärvi & Hurme, 2015, p. 48). Semi-structured interview was chosen to maintain a relevant scope during the interviews while also allowing interesting details the surface by digging deeper to interesting aspects with on-the-spot follow-up questions. In more detail, a semi-structured method called a thematic interview was utilised to scope interview topics to three distinctive themes and allow customisation within them. This is a benefit of thematic interviews, which Hirsjärvi & Hurme (2015, p. 48) describe to promote the voice of the interviewee in data.

Interview questions were shifted to reveal details from the perspective of the interviewee's position in the development model in addition to general level by wording them to highlight the interviewee's own experiences and knowledge which is a key characteristic of thematic interview style (Hirsjärvi & Hurme, 2015, p. 48). Particular attention was also paid to creating questions which could not be answered with a simple "yes" or "no" to collect more meaningful views. Questions were categorised to following three main themes based on their type and which research question they were targeted towards:

Theme 0: Background information
Theme 1: Challenges of the scaled agile method and configuration
- RQ1.1

Theme 2: Customer value in the current model and prioritisation
- RQ2.1 and RQ2.2

The full body of the English semi-structured interview is visible in Appendix 2 and the Finnish equivalent in Appendix 3. The questions were more targeted and detailed than generally in semi-structured interviews with interviewer's familiarity of the target area by having worked in the organisation and applicable findings of the literature review. This more specific wording was expected to increase comparability of answers while still allowing studying the interviewees' perceptions of the topics without overly guiding them. The interviews still had plenty of room for modifications on a per-situation basis to support the interview situation and context of points that had been discussed before each question. This promoted the role of the interviewer also in charge of the interviews by asking follow-up questions of relevant topics that were seen as potential empirical material, typical and fruitful for semi-structured interviews (Hirsjärvi & Hurme, 2015, p. 66).

In the interview body, Theme 0 serves as brief background information of the interview to grasp the position and experience of the interviewees to support validity of the results by highlighting the years of experience and the representativeness of the gathered interviewees by their level. Theme 1 is pointed towards research question 1.1 to reveal how the interviewees see the current configuration of SAFe to work in and around their position. Theme 2 aims to reveal details to the value-oriented research questions 2.1 and 2.2 by focusing on the current meaning and status of value in the development model and how they could be improved.

The interviewed personnel consist of employees who work in various positions in the development model. The selection of these interviewees was performed together with the instructor from target organisation to simultaneously gather a diverse and representative sample of personnel and their knowledge and experiences. SAFe's levels in the organisation (team, program, portfolio) were used as guiding elements to ensure representativeness by inviting a balanced number of experts from each of these levels. This was done to form a heterogenous set of responses from various positions in the bigger picture. The selection of interviewees was limited to certain internal segment of the company to inspect a specific context of SAFe configuration in the organisation as there are differences between them.

6.3 Interview process

Potential interview candidates were sent introductory emails as invitations to the study and if they answered willingly, they were invited to an interview meeting. Ultimately, eight interviewees were acquired plus the instructor of the thesis from the firm's side as a test interview before moving on to the invitees. The test interview formed a baseline for the length, possible challenges and overall guidance that could be necessary later. The interviewees were sent the interview body to allow familiarisation and gathering thoughts around the topics in advance. This was thought to potentially yield more concise and thorough answers. It was however noted in the invitation email, that answers were not expected to be prepared in advance and that the nature of the interview situation would resemble a discussion where the interview body would serve as a structural and thematical guide.

All the interviews were performed individually with Microsoft Teams online meeting software during January and February of 2023. Teams is the primary means for all meetings in the target organisation and therefore formed a natural and effortless platform for interviews. Teams was also used to record and transcribe the interviews with the permission of the interviewees. Recordings were used to revisit the interviews at later time to verify responses and collect additional details which could have gone unnoticed otherwise. Recordings were stored privately in Microsoft Stream according to the organisation's

strict data handling policies. Only the meeting participants, the interviewer and the interviewee had access to the recording and initial transcription.

All except one interview were held in Finnish as it was the first language of both the interviewer and interviewee. If this was not the case, the interview was performed in English. One interview had to be continued in the following day due to an urgent matter the interviewee had to attend. Transcription of the interviews was done in the language the interview was conducted in, and their findings were translated to English in the analysis if used as a quote. As built-in automated transcription of Teams was used, its result was reviewed as soon as possible after the interview to correct mistakes and redact sensitive information of the company and interviewees. The interviewer also took private notes during the interview to support later analysis. When the interviews had been held, their amount (8) was seen sufficient as many common topics had already emerged with similar patterns and characteristics even from participants on different levels and few new topics surfacing towards the last interviews.

6.4 Analysis of the interview data

Qualitative data is often described as both a blessing and a curse. On one hand, explaining things in words often yields descriptive results but on the other, the relative effortlessness of collecting this type of data may easily consume significant resources in its appropriate analysis (Robson & McCartan, 2016, pp. 459–460). As expected, this challenge also realised in this study as the interviews lasted between a bit less than an hour to around 1,5 hours and produced 17-25 pages of pure textual transcription material. The lengths of the interviews are listed in Table 3.

Table 3 Lengths of the conducted interviews

Interviewee	**Interview length**
Interviewee 1	52min 29s
Interviewee 2	1h 11min 36s
Interviewee 3	1h 24min 38s
Interviewee 4	1h 21min 12s
Interviewee 5	1h 3min 55s
Interviewee 6	1h 22min 52s
Interviewee 7	1h 14min 41s
Interviewee 8	1h 10min 31s

Robson & McCartan (2016, p. 260) call analysis the central challenge of quantitative data, as its agreed methods are scarce. Thematic analysis method was chosen for this study as it allows inspecting collected data flexibly. It is a common qualitative method which originates from psychology but is seen as widely adaptable to other fields as well. Its main goal is to recognise, analyse and report models (themes) unambiguously by finding and recognising patterns that

share a "meaning" from the empirical data. (Braun & Clarke, 2006) DeSantis and Ugarriza (2000) describe a theme to be an "abstract entity that brings meaning and identity to a recurrent experience" to support qualitative data analysis.

Under a broader term of content analysis, under which thematic analysis of qualitative can be categorised, Hsieh & Shannon (2005) identify three main approaches: conventional, directional, and summative content analysis. The analysis of the acquired textual interview data of this thesis strongly resembles the nature of conventional content analysis, which heavily leans on observation as the starting point for identifying codes (or themes in the sense of thematic analysis). Still, a connection to directed analysis, where codes root from existing research (Hsieh & Shannon, 2005) can also be identified, as findings from the earlier literature review can be seen to affect the thinking of otherwise conventional content analysis of textual material.

Thematic analysis also possesses its own benefits and disadvantages. As mentioned before, its flexibility allows it to be utilised in various contexts, such as with different kind and sort of data. It also does not require extensive experience of qualitative methods in general and is therefore quick to get a grip of. Thematic analysis is particularly suitable for researching views of diverse subjects, their commonalities, and differences as well as summarising findings from even large amounts of data in a concise form. (Braun & Clarke, 2006)

These benefits also lead to some of thematic analysis' downsides as a qualitative method. As the method does not require very much experience from the researcher, this may lead to lack of rigor: trust or confidence in the analysis. Also, use of language is not a subject of interest in thematic analysis and therefore its potential findings are left uncovered. (Braun & Clarke, 2006) The same writers list the phases of thematic analysis to consist of:

1. Getting to know your data
2. Creating preliminary codes
3. Finding themes
4. Reviewing themes
5. Naming and defining themes
6. Producing the report

The previous phases were mostly followed in the data analysis, which was started in parallel with the last interviews. This is possible and common in qualitative research (Braun & Clarke, 2006; Hirsjärvi & Hurme, 2015). The analysis of the interviews started with verifying the transcriptions which had been produced during the interviews. Corrections to increase the readability and clarity of the transcriptions had to be made, which also provided a chance to revisit the interviews in detail by listening to them once again. This allowed identifying topics of interest and preliminary themes among the interviews as well as collecting grades and other potential remarks for later use. This roughly resembled the three-fold process of preliminary empirical data analysis identified by Hirsjärvi et al., (2009):

1. Verification
2. Complementation
3. Organisation

Commonalities and contradicting viewpoints were collected as notes with other answers providing more depth to them. These notes were first arranged according to the order of the interview body, and later separated to themes which they had common outside the interview themes. Hirsjärvi and Hurme (2015, p. 173) call this "theming", where the initial themes of the interview are mixed and complemented with ones surfacing from the interviewees' responses. This also resembles the third and fourth step, finding and reviewing themes, listed by Braun and Clarke (2006).

The identified themes had also been assigned preliminary names at this point to clearly distinguish them from others. In accordance with the method, the identified themes were not limited to ones in the interview questions but identified on a blank sheet basis by comparing the interview contents across its questions and themes. Finally in the sixth stage, the themes were brought over to the final document and complemented with direct quotes from the interviews, which had been marked earlier in familiarisation of data. Themes were laid out in primary themes and their more detailed subthemes to increase readability. Also, averages and medians of the grades the interviewees gave were calculated and brought over in visual form to themes they were seen to provide relevant information to.

7 DESCRIPTION OF THE TARGET COMPANY AND INTERVIEWEES

Before presenting the empiric findings it is relevant to introduce the interview context: the commissioning company of this thesis in more detail. This section is dedicated to doing that with also detailing how the subject of this thesis came to be through personal observations during a period of employment. Background information of the selected and participating interviewees is also provided to gain understanding of their nature and relevance.

7.1 Description of the target company and roots of the subject

The company which serves as the commissioner and context of this thesis is a major telecommunications service provider in Finland and has operations in other Nordic and Baltic markets as well. It has long roots in the Finnish market and currently operates in B2C, B2B and Business-to-Operator market segments. The company has over 3000 employees in Finland alone, and heavily subcontracts work from external service providers on customer service call centres and IT services and development, for example.

Thanks to the company's established nature and competitive offerings it is one of the key players in the Finnish very telecommunications market. In terms of B2B customers, the development segment where this thesis concentrates on, the company serves customers ranging from small and medium companies (SMEs) to the largest public and private organisations in Finland and even globally. The extremely tight competition in the Finnish telco market together with heavy regulation shape the company's operations, which despite of its incumbent status, is in continuous motion to respond to market and shareholder needs by maintaining competitiveness and increasing efficiency. One of the most important focuses for the company is to provide superior customer experience through understanding the customer and new technology as differentiat-

ing factors to gain competitive advantage, typical for telcos of today (Penkert et al., 2019).

The company has adopted SAFe as its scaled agile development framework of choice several years ago and has since created its own way of working around it, which still heavily resembles SAFe's vanilla configuration. The Finnish country unit was the first internal segment to pursue and implement SAFe. The SAFe-based way of working is being implemented in other countries as well to unify working methods. The company has named its own development model as [the company's name] Way of Working, which includes customisations that have been created during the years of maturation. Perhaps the most significant of these is the absence of the large solution level of SAFe's structural layers.

The line organisation in the company has its separate structure from the development model and is largely organised to country divisions. There are also common group-level units with house technologies and support services which are shared with country divisions and therefore provided commonly. The country divisions follow a roughly business segmental structure while individual departments for country specific needs, such as infrastructure and development activities are also in place. This thesis concentrates on B2B development in Finland to provide a defined context while still spanning across many functions across the line organisation. Exact structures and processes differ across the line organisation with some functions having their own, but generally minor, adaptations in the way of working.

The development activities are located in various functions around the organisation depending on the topic. In Finland, there is a centralised development department which houses experts such as analysts, process experts, (product) modellers, service designers and many other roles leaning on the portfolio and program levels of SAFe. Geographically, the majority of technical development is done in the company's headquarters, but specialists can still be located in other Finnish offices and even abroad in some cases. Technical experts of specific systems or platforms are often situated on the common group-level organisation, from where they pitch in to development in their respective parts. The SAFe train configuration is largely structured on a business segmental basis to support, say B2B development. The teams inside these trains are structured on a per solution or system basis to support their development, such as the CRM platform, online portals, integrations, et cetera.

The spark for the subject of thesis ignited during my employment period in the company's Finnish development department. In a training session of the current way of working model, questions arose of the current model's rigid and inflexible nature which makes it difficult to implement minor changes and new development quickly, which could yield exponential customer improvements. I was inspired by this idea and began discussions of inspecting the current model by writing a master's thesis of its challenges in the light of scaled agile literature, which upon initial searches, proved to form a solid foundation to build on.

I took up to the topic with my previous manager, who agreed with the discussed issues, saw the potential in the work and directed me to contact one

of the agile coaches of the company in Finland. He also was also extremely interested in digging deeper to the current state of the company's model and provided extremely useful background information of the history, issues and the actions that had been used to tweak the model over the years. He also agreed with the practical angles and agreed to serve as the instructor of practice of this thesis, while my previous manager handled the needed agreements and other details. Over the course of several meetings with the company's instructor to find the most potential practical angle to the subject and university side supervisor to balance this angle with academic requirements and guidelines, the thesis was set on the track of reflecting challenges of the company's model to ones identifiable from scaled agile literature alongside a strong focus on researching the role of customer value as means for optimal decision making in development.

Alongside serving as the context of practice for the interviews, the commissioning company offered tools (laptop, phone, other utilities) and access to internal systems to help the writing process and setting up interviews. Internal documentation was not used to provide additional insight. Therefore, all discussed internal details of the company are based on meetings that have occurred with the internal instructor, previous manager, and the writer's own recollections from the employment period. The interviews of the next section also offer objective and diverse viewpoints of the company's details. All information has been verified to ensure confidentiality requirements the company has set for the work.

7.2 Background information of the interviewees

The interviewed personnel all had multiple years of experience from working around agile methods, ranging from a minimum of 4 years to an interviewee describing their experience to root even to the days of the agile manifesto. The interviewees made distinctions on their experience of team-level methods, mainly Scrum and the Scaled Agile Framework. The experience of SAFe was generally shorter than experience of agile overall although some interviewees described SAFe as their first touch to agility in practice. Generally, the experience had been acquired from the host organisation with a few interviewees describing having experience of agile methods from other organisations as well.

The positions of the interviewees in the SAFe model distributed quite evenly on all levels, where the lack of large solution level in the organisation, a customisation of the implementation, must be remembered. Portfolio level was the most common level of the participants with the team level having the fewest participants with two interviewees. Some also described their role to span across multiple levels with their primary level being included in Table 4 of the characteristics of the interviewees. An interviewee on portfolio level also described their role to favour business side, an additional desirable viewpoint for the study. Developers were not included as interviewees, because in the organi-

sation they are often contractors from external suppliers and therefore may not have the required experience and/or the lens of the target organisation which were sought after when seeking interviewees. All interviewees have strong experience of working around agile methods and their diverse nature support the reliability of results.

Table 4 Interviewees' levels in SAFe and years of experience of agile

Interviewee	**Level in SAFe**	**Experience of agile in years**
Interviewee 1	Program	5
Interviewee 2	Program/portfolio	10
Interviewee 3	Portfolio	4
Interviewee 4	Portfolio/business	18
Interviewee 5	Program/business	15
Interviewee 6	Portfolio	6
Interviewee 7	Team	5
Interviewee 8	Team/other	7

8 FINDINGS

This section explains the interview findings to answer the research questions of thesis from empirical material. First, general perceptions are provided of the current model, which is followed by the identified challenges, and finally customer value's role and measurement in the development model. The answers have been categorised to themes according to thematic analysis method described in previous methodology chapter. There are four main themes, one of which has more specific subthemes to enhance readability and promote their distinct nature.

The points under every theme are first explained at a general level and supported by direct quotes from the interviewees to bring out their voice regarding them. Each quote is followed by the number of the interviewee which demonstrates representative use of all the interviews in a balanced manner. Their levels in SAFe are also mentioned where they are considered to provide additional insight. The levels only reflect to the interviewee's position in SAFe with words such as higher and lower having no implication to their ranking or quality in any way. This section serves as a backbone for later discussion with literature and other interesting aspects before implications.

8.1 Perceptions of SAFe and the current model

The overall perception of SAFe was positive with its benefits seen to cover its downsides. The development model itself was seen to have continuously improved from its early days of adoption, with eagerness towards further iteration being also a clear ambition. One interviewee had experience from working with SAFe in another organisation and based on that considered that the current organisation's model had reached a much more mature state and therefore could support teams better.

> Really useful for this company for the project that they have it right now here in this [organisation]. I found it that is really reliable tools that we have it right now and we

> can just manage the teams, can manage the products and the resources here. (Interviewee 7)

Many interviewees considered the structure that SAFe provides a very welcome addition and clear benefit to the organisation together with cadence, collaboration, clear prioritisation, and transparency which were also seen as definitive benefits of the current model. Only two of the interviewees assigned a less-than-average score for it in terms of a scale from 0 to 5, where 5 is the best. The average grade of the interviewees was 3,13 and the median 3,25.

> I like this way of working and as it is a sort of a framework which everyone adapts to their best and we have our own way of working in [the organisation] while trying to follow SAFe's guidelines and I think it works really well. (Interviewee 1)

> General perception is mainly positive, because it provides us with the cadence and scheduling and the expectations of that schedule how we have to do things. So, it creates us the structure of having certain things four times a year and deadlines. (Interviewee 2)

> I find SAFe to be one of the best [models] which I have explored for continuous delivery and execution. (Interviewee 3)

> Really great framework, which in a way brings common ways of work, cadences, ceremonies, and creates a common method -- how we get things to work. -- We have a clear way of getting things to move forward and clear roles and endpoints. (Interviewee 6)

Interviewees connected the positive remarks of SAFe in general to apply to the organisation's implementation as well. In a more critical view, an interviewee described SAFe as a generally acceptable framework but called the organisation's adaptation to lack a human aspect. They also added how iterating the development model itself was not up to SAFe's standards in the organisation which suffered from locked-in-place, personified responsibilities and other effects which can be viewed as downsides of earlier structural benefits in the form of bureaucracy. They added how they did not know whether these and the incapability to change were SAFe's or the organisation's fault. Another interviewee also presented similar views regarding scatteredness of roles and communication between them. On the upside, development items and their prioritisation were perceived to clearly link to the strategy of the company. They also considered that problems are openly discussed and communicated and felt what there was strong commitment and support for the development model.

> Let's put it this way that it brings certain kind of bureaucracy to the machine and kinda blocks people from communicating with each other. At least currently it is sort of a ticketing machine and [so] offers visibility to higher levels on what kind of development is done in the whole organisation. (Interviewee 5)

> If we don't have a holistic understanding or will, it can quickly direct us to have specific roles and taking care of [only] your own part –- doing your own piece and handing it over to the next etc., does all [knowledge] transfer in reality? (Interviewee 6)

> Here at [the company] we have strong support from management, and we have also been kinda organised how it supports or way of working which makes development much easier in that sense. (Interviewee 1)

One interviewee considered changes in the line organisation, which had been performed about a year ago, to have influenced the working culture substantially by mixing people with different backgrounds and working culture together. They described this in a positive light, with some friction still causing issues but felt that things were still improving with time. Several interviewees also felt that the current model is inflexible and very slow when it comes to the actual throughput of development items.

> In the latest change in the organisation, we brought -- as previously we had roles purely around the train, and now we have [roles] from various backgrounds here. (Interviewee 6)

> A very large funnel beginning with small ending, where the throughput is small. A lot of stuff what we want to develop is put in, but very little comes out. (Interviewee 5)

> At times it feels like we are shooting a fly with a cannon. (Interviewee 4)

> You hear feedback from outside a development train that it is seen as a pipe where things are thrown for development, they progress and come out sometime in the future. And the perception I have from SAFe does not support this, which is why we should be able to do more. (Interviewee 6)

In a contradictory view another interviewee with more experience of working around agile and previous methods in the organisation listed increased amount of actual throughput a concrete and measured benefit of the current model compared to previous. However, they had a clear distinction of what benefits were thanks to SAFe and could be achieved with other agile methods. Then again, they also considered that the current model is highly predictable and measuring works well. Overall, despite its clear challenges, they did not see practical alternatives by highlighting the history with waterfall methods and alternative development funding models they had witnessed.

> Factually, and when measured, we get more things done -- but I don't know if there are any other good things. Or all the rest of the good things we could solve some other way but there are indirect benefits when we use a modern development method -- such as attracting talent -- in 2023 no self-respecting developer works with any other method than agile. (Interviewee 4)

Interviewees called Finland to lead the implementation and adaptation of SAFe of all the countries in the organisation operates in, which has also led to some matureness of the model which is still finding its place or being implemented in other countries. Still, two interviewees wondered if people had the required knowledge and understanding of SAFe and the organisation's customisations.

> It would ease things if people understood our way of working how it works, how prioritisation works, etc. (Interviewee 3)

> -- I guess that understanding of SAFe is still partly lacking, that people would understand cadence, and that we have [organisation's] development model, but what it means in practice is another thing. (Interviewee 2)

The issue of understanding the difference between a tool, such as one for ticketing in the organisation, and actual method or framework was also mentioned by one interviewee, who pondered whether people understood their difference.

> Do people understand that an agile development methodology, regardless of the model, is not the same as the used ticketing system? (Interviewee 2)

8.2 Challenges

There were several distinct themes of challenges, which repeated in the interviews, which are now explained in their own subsections. These challenges provide answers to research question 1.1 of this thesis. Key challenges which surfaced in the interviews were early over-analysis and solution centricity, lack of customer input and feedback, insufficient data quality and measurability, and challenges related to resourcing.

8.2.1 Structural over-analysis and solution centricity

A very commonly recognised challenge by the interviewees was that development items are analysed too far in the preliminary stages, which leads to solution-oriented development. Heavy early analysis locks the idea of a solution in place and the development model was therefore perceived as very solution oriented. The interviewees considered that development would benefit of thorough problem analysis by different collaborating experts, which could reveal required details before the solution is developed and changes to it become increasingly costly and difficult to implement. One interviewee went as far as to call this early over-analysis to have a paralysing effect to development and added how steps had been taken towards proper problem analysis in the form of pre-analysis of epics before they progress. Overall, this challenge was seen to limit team autonomy severely and increase the distance between customers and solutions.

> Analysis should not turn into a paralysis. – We are more around the solution than the actual customer need, which should be the one guiding development more. -- Oftentimes we start solving things -- by building new capabilities on the gut feeling what feels like the best solution in the beginning. Pre-epic work would increase our understanding of the problem in our environment, which would make it easier to select – you know, be it a hatchet, shovel – the right tool. The right way to start solving the problem. (Interviewee 6)

> At the moment we plan very much on the behalf of the team. Little room is left for the team in our current configuration. -- We have driven our configuration to a model where we analyse things very far and define what we want to achieve from all possible angles. Therefore, the teams are not presented with a problem to solve but rather a complete specification to type into code. (Interviewee 4)

> In my opinion everything comes pretty much as given -- in a very chewed form to the team, I don't really see any ability [for teams] to make decisions. – I'd like to hear the team's voice more in development, but in our model, it does not appear at all. (Interviewee 5)

Interviewees also felt that over-analysis and solution orientation did not affect all aspects of development and that teams had room to decide things in some regards such as in terms of technical details. An interviewee on team level described teams to have certain freedom of work where they are not interfered with if the development items are produced in schedule. However, the definition of team-level autonomy was also questioned by two interviewees, who after consideration were not sure where they would draw the line in terms of the level of truly agile decision-making.

> So, in practice where team can make decisions – is technical solutions. Our goal in analysis is to describe use cases with personas or equivalent but team is the one who sees how it will be built and done in technical terms which is where the teams should definitively have autonomy. -- the autonomy is bigger because we [on higher levels] are not interested in how it is done in technical terms as long as it works. (Interviewee 2)

> They [managers on higher levels] are not really interfering about things and the details of our routine work so they are letting us do our work as long as it's okay and we are delivering. [When it comes to] inside the team how we are developing and how we are doing our implementation or just the technical details, I really don't see any kind of the control or being monitored by the managers. (Interviewee 7)

> -- not micromanaging what teams do but trusting that when we give and bring goals in accordance with our strategy what has to my achieved by the machine, I feel that at [organisation] we give space quite well [for the teams]. (Interviewee 3)

> For me, team level autonomy does not mean the team can come up with anything – rather, the team is told that here's the swim lane, there are its borders and there far away you can see the ceramic tiles in the end which we do not know how to reach but you should stay on the lane and reach the tiles. Maybe team-level autonomy is the power to make solutions, based on expert evaluation, for example that we should do this [development] to the old system as it makes [migrating] things easier because a new system is coming. (Interviewee 4)

Interestingly, two interviewees also explained how team-level decision making and autonomy might not even be desirable for the organisation due to the nature of development resources, firstly from perspective of organisational autonomy and secondly from cultural factors. Ready-made specifications were seen to discourage developers from showing initiative and bringing forwards their

views. Still, interviewees weren't sure whether increased autonomy would actually benefit things with many finding the current alignment between planning and team autonomy quite suitable for the organisation.

> But we cannot give a certain amount of autonomy because we have many teams which are purely managed by our suppliers and consulting firms, and we do not want to hand over authority from [organisation] to external corporations on the content what we are doing. -- Moreover, I see that we don't get ideas from teams on what they would like to improve, and we have quite a lot of resources from [an Asian country] – and to put it bluntly, the level of initiative there does not quite match for example Finland or Estonia. (Interviewee 2)

> Perhaps the best combination of this is that we have work management in a country where liberal [work ethic] is a virtue and our developers come from -- hierarchical societies which is also their way of working. And when we subcontract work [from these environments] it relies on what is asked, and that is pretty much what is ultimately done and delivered. (Interviewee 3)

> And it is a self-feeding organism in way that if teams are taught to being told and given everything, creativity will not flourish. -- To be honest, I cannot estimate if we would perform better if teams could decide more and planning wasn't as dominant [as it currently is]. (Interviewee 4)

When asked about grading the status regarding team-level autonomy and forward planning, the interviewees' views were quite mixed with the median grade being 2,5 out of 5. Interestingly, the interviewees on portfolio level rated autonomy a bit higher than interviewees on lower levels. This can be seen from the array chart in Figure 3. The dimensions present individual interviewees with their level and grade present in the visualisation.

Figure 3 Interviewees' grades on planning vs. team-level autonomy in current configuration from 0 (planning) to 5 (autonomy)

As a possible remedy to shift analysis towards the problem from solution, an interviewee added how they had tried to incorporate elements from design science to the model. Still, they considered that structural elements were in the way of better working in the form of too many managerial layers and mismatched train configuration in terms of value streams. They added that they would like to see smaller development units from the current segment-based

separation to increase comparability of development. This was an aspect another interviewee mentioned as well.

8.2.2 Impractical comparability and measurability between functions and value streams

One of the central concepts of SAFe is to build coordinating agile teams along the organisation's value streams (Knaster & Leffingwell, 2020, pp. 135–137). Three separate interviewees on the lower levels pointed out severe misconfiguration regarding organising development around streams of value, as currently trains and teams are built to serve a particular business segment or other function, such as a system or solution.

> Streams do not flow – streams have been separated to different containers [contents of] which we try to pour into the next one: the containers are different size, so [the contents] don't fit and therefore don't flow. (Interviewee 8)

Another interviewee along the same topic found many challenges to root from the fear of wasting resources to run smaller functions and development instead of a large and centralised functions. They considered that separating development to more trains, for example, would help in prioritising more comparable issues more effectively and bring other benefits as well, which currently are not recognised in the organisation. They said that this way development would be more aligned to actual value streams in the organisation where experts could focus and collaborate on things that they have more expertise on. The same interviewee also would like to see certain responsibilities shifted from being centralised to individual roles to more discussion-based approach.

> In our approach, a whole train is utilised by a segment which means, that there are really many value streams. -- Of course, we are producing stuff to all [specific customers] but it is such a broad term that I think that we should just cut and separate [development] to smaller value streams and sensible entities where things are if not jointly measurable, at least jointly discussable so that there wouldn't be fundamentally different items being prioritised or enumerated. (Interviewee 5)

The beforementioned structural challenges in team and train configuration and the issue of difficult, impossible, or at least undesirable comparison of development functions was one of the most common issues mentioned by other interviewees as well.

> We have [internal tool] configuration work, which is more technical. And then we have another sort of thing that we make products. We make functionalities to customer service, so measuring [these] is challenging. And right now, we try to put everything to this kind of funnel and that is where we currently face issues. – These trains can't be measured similarly and right now our management sees them as similar. They are measured similarly when in reality it does not work this way. (Interviewee 1)

Many mentioned the following comment regarding measurability often word by word: "You get what you measure". Reliable and effective measurability was seen as a prominent challenge leading to difficulties in improving operations and prioritisation. An interviewee working closely in high-level prioritisation deemed data quality to be one major factor limitation in effective data-driven decisions due to a migration to common ticketing platform, which lacked useful fields that were present in the previous tool.

> -- another challenging thing is that when we migrated [our ticketing tool] to a new one, the previous one had fields such as for business case, this is [x amount of money]. -- They were easy to export to Excel and process there. Now we don't have those fields anymore because they were tailor-made. -- And currently it extremely cumbersome get them to Excel. (Interviewee 3)

One interviewee also highlighted transparency along a value stream to suffer from locked responsibilities where individual tasks and contributions remain separated in different parts of the organisation. This leads to the big picture being indefinite and unsynchronised in their view and value delivery suffering severely, resting on the responsibility of managers to complete.

> So that it would become somehow clearer that we are not only solving tickets or a specific amount of [other tasks] but would instead see more clearly what the customer impact is – this is such a big structure – that I see that we have some difficulties when it comes to people seeing the whole chain. This leads to [people] solving the part of their own system or team which often is not the complete chain and depending on the quality of [their] manager, is left to gather rest of the troops and communicating the whole chain. (Interviewee 8)

A separate interviewee presented similar remarks through their experience how they had been very confused how teams in a train relate on one another when they started to work in the organisation. Also, they said that teams which were not really working among same topics were sometimes forced to attend meetings which were not useful for them, but this has since improved. Overall, the interviewee did not explicitly mention value streams, but reflected on a similar point how teams with different duties and themes worked in the same train which did not have a shared goal. Instead, their uniting factor was the business segment or solution their work was produced to.

> …it was awful to find my way in this train -- I didn't know how this team was connected to other teams; what other teams actually do... I could hear quite many names, quite many complicated names I had never heard in my life, and it was really, really challenging to find out the relationship between the teams here. And when it comes to the [documentation tool] for example, the one place to document all information about the whole train was also such a mess. (Interviewee 7)

Another interviewee considered comparability challenges to hinder prioritisation activities by making ranking development items to one order difficult by diluting their characteristics under one scoring.

> If prioritisation is that we rank items based in their importance, I feel it works really well. If prioritisation is that we really choose and decide that ok, our capacity can take let's say 20 things this year and all the rest is wiped out – all the rest is just noise which we won't concentrate on: then it doesn't work. – if you have 100 things on a list, the difference between the 89th and 85th is diluted. (Interviewee 4)

8.2.3 Unpredictable and predefined resourcing

Three interviewees pointed out challenges regarding the nature how the company acquires developers turn the development items to working solutions in terms of code. The developers generally come from contractors as consultants, who work in the company's team for a fixed period at a time. Consultants, or developers in practice, can also come from different providers. One interviewee felt that these characteristics limit the technological viewpoints and knowhow which could be leveraged for innovation, as currently development teams are gathered based on existing technology stacks. This problem does not directly relate to SAFe itself, but how development work is funded in current model.

> We have an enormous amount of technology: we are a big house so name a technology and we have it, which means that building teams and therefore managing investments becomes difficult. What I mean by this is that when we allocate investments case to say strategical themes, -- the flow of cash begins to shrink smaller and smaller. And then within development investment cases the money is allocated to teams so in practice we buy capacity for half a year or year. -- Well, this capacity is of course built around the technology that we have there and therefore the competence of the team is quite limited. -- so, when development funnel spits out problems, they are solved with our existing technologies and knowledge and no great innovation can occur. (Interviewee 4)

Two other interviewees revealed major difficulties regarding unpredictability to root from temporary and uncertain nature of development resources. Developer turnover from their own will or vendor consolidation was seen to create too much change and so affect the atmosphere in development teams very negatively and decrease their overall output and efficiency. Turnover also created the need for new people to learn the company's ways and solutions and removed the acquired silent knowledge of more experienced contractors.

> Once they decided that, okay, for some reason they don't like to work with current vendors that we had at the time. And then it was a sudden change and they let some of our more senior [developers] goes go away. And it was really sudden change. And it affected the team a lot and the whole PI delivery just failed at once. And it was because of those things that they did that we -- we couldn't do anything with because it was out of our hands. -- So, we had to let our features away and we told them [management], okay, so we don't have resources, we don't have any delivery for the features [of this PI].
>
> I just gave [upper management] one example in one meeting: what you are doing is -- is that you are [creating a situation] that you have a flight on the sky with the passengers and the whole crew and the pilots and you are [forcing] the most senior ones [to] go out of the flight in the middle of sky. And that means that you are let-

> ting the pilots and the flight crew, you know, jump from the flight and you have now the flight in the sky to the destination with the passengers. No pilot in it. And you are just saying to us. Yeah. OK, let's hope someone among the passengers knows or is brave enough to take care of this plane. That's what exactly happened to [a certain] team. That's why [people] are having little bit challenges with that flight to get to destination even now. (Interviewee 7)

> I see that coding is very creative work. – and to be creative, you need to feel safe. – And at the moment I don't think that the feeling it quite safe in our teams. What causes this, it may be the turnover which is our own fault. Vendor consolidation is practiced, and this may be why the team members don't feel the safest in the world. And we have to turnover, and we should establish some sort of peace in the teams and certainty that there will be work and so we could maybe get them to stay for longer, and then they would have better knowledge and stronger hold of their plot and role. (Interviewee 8)

Interviewee of the last paragraph and another on a higher level also described decreasing internal resources and increasing amount of work to create never-ending rush and decrease the quality of work with development items piling up on backlogs. Also, the first of these interviewees felt that performing both legacy and transformative development simultaneously eats up a lot of resources as they require unique approaches.

> And we cannot fulfil the customer expectations optimally, because in my opinion, one value for the customer is that they receive their development work in a reasonable amount of time. (Interviewee 3)

> Currently we develop our legacy systems and progress our transformation, so no one has time to focus solely on one thing -- we all run too fast to both directions which leads to not having the time to thoroughly think. And it isn't anyone's fault but certainly reflects negatively. -- Fewer people, but the feast does not get any better. The pressure builds more and more on the remaining and when you want to do things well you notice your insufficiency when there just isn't enough time. (Interviewee 8)

8.2.4 Insufficient customer input and feedback

Perhaps the most common challenge, mentioned by almost every interviewee, in the current development model was the lack of customer input from internal customers and feedback in it. While there are measures in place to incorporate feedback to the current model, its amount and methods were not seen sufficient. Many also called there never to be enough customer feedback and input as it was seen extremely beneficial with many practical effects. This was a unified message of almost all interviewees. From the following answers from higher levels, it became clear how this problem was seen to particularly affect them. Some aspects of this theme touch on the challenges described earlier, but still form an entity of its own.

> There's too little voice of the experts from the target organisation as well as strong guidance from business side on which direction we are headed to. Therefore, we al-

> ways have the risk – in a development train – to progress through IT-development [which limits viewpoints]. I'd like to involve our end-users and others in early stages to specify requirements. Here [on higher levels], development specialists create a lot of requirements, and they are somewhat aware of the day-to-day operations in customer service. But sometimes we get feedback that development specialists create stuff, we test it, hear end-users, and get additional requirements from them. We start the process again or go live and get feedback why this kind of stuff is brought here, why is this this way and that that way. (Interviewee 2)

> We get a lot of requirements, creation of which has not involved end-users. They have been enriching requirements and possibly will test the ones that are developed. – I'd hope that these end users would be specifying things to be aware what is coming and what they will eventually test. [if this isn't done] expectations can be something else entirely. (Interviewee 1)

Interestingly, there was only one interviewee who was close to being satisfied with the amount of customer input around their role. This interviewee worked on team level, where they had concrete methods to increase awareness of customer's day to day activities. Overall, customer input was seen extremely valuable to grasp the reality of customer demands and do the right things, which sometimes can even be less work than initially thought.

> -- we are presenting some kind of the analysis of the market to our developers every week and how the and the customer reaction to our [system] and product and it's called feedback from the customers. [asked if there was a sufficient amount of customer voice] I believe that you never can say there's enough. Or that it's perfect. But we are doing our best to get enough feedback. -- We are getting the feedback from the audience and mostly this feedback is coming from the business and marketing. It is so, so useful and I can say valuable for the teams, and the developers. They get the ideas to go and move forward based off those you know. -- Sometimes we had comments that even helped us to reduce our, you know the workload because we noticed OK, we made some mistake, maybe we shouldn't go through this path. So, we can just do less work and then still have happy marketing people and satisfied customers and I really found it really valuable. (Interviewee 7)

Defining specifications and solutions based on a false understanding of customer demands and wishes creates wrong kinds of solutions. If development is extensively analysed and starts to get onto these wrong tracks in the beginning of development, changing the solution based on the feedback from a demo event, for example, is difficult.

> This is our repeating sin of sorts; we still aim to develop [items] to too complete state. – And as the result, the customer does not receive work from the initial stages – and we don't get to test the hypothesis early enough and secondly, the customer does not receive the development [result] quickly enough. (Interviewee 3)

Multiple interviewees promoted the vitality and benefits of acquiring specialists familiar with customer's needs, such as customer service agents, to testing activities. However, the interviewees recognised severe challenges in terms of at-

tracting these volunteers from operative positions. They called for a need for new methods to attract diverse and insightful testers, as currently the operative staff did not have incentives in place to participate in testing from their own duties, other than their own interest, which was not guaranteed.

> Our [internal] organisations are not very willing to give these people to these [testing] purposes. And of course, the people themselves must also be willing, because we have people who are on a provision-based salary model. There are sales bonuses in customer service etc. So, if we want to contribute people in this, common good [activity] of sort, we should somehow compensate the possible loss [of income] to them. (Interviewee 2)

> It can be someone from sales, someone from anywhere or billing. We have challenges to attract these people to test. I understand customer service for example, if you are not incentivised for testing it is all out of your compensation when you are not closing deals. There should be some sort of motivating being done here. (Interviewee 1)

Two interviewees also identified challenges in the form of feedback that is currently measured. They said that feedback in terms in numbers and grades is easy to get from customers but does not really provide constructive feedback on what to actually improve. Also, the difficulty of pinpointing received feedback to the relevant part of the customer's journey was not always successful, which led to the source of feedback and its actual target remaining unclear.

> Not every customer is giving us feedback. You know we have two different ways of receiving the feedback: by commenting or just giving us some numbers. So, most people don't like to write comments, but they like to just give the number you know. -- We receive quite a lot [of numerical feedback] but we don't know the reasons [behind the numbers]. Could you just, you know, elaborate some explanation? (Interviewee 7)

> Like a CSAT survey, for example, which comes up in a specific touchpoint without a context. And in our business, customer processes are oftentimes long, they are phased, and it would be interesting if we could monitor the customer paths from start to end. -- If we had the measuring points during the journey, then we could follow up on the creation of customer value a lot more efficiently and precisely in specific parts of the process. If we receive an overall grade of 3,5 from the process, we can tap ourselves on the back. But if we notice that the inspiration phase was really hard, grade 2 for example, we know that we must do something to make it better. (Interviewee 4)

In addition to attracting more and better testers from operative positions and receiving more constructive feedback, interviewees came up with several possible remedies to lacking customer input. The most common actions were centred around promoting activities such as demos, visiting the company's retail stores, and listening to customer service calls to grasp the day-to-day of customers more concretely on higher levels as well.

> A thing we used to have but has now been missing is these demo events, where what's done is demoed. Customer should be able to tell that wait a minute, this is an especially important thing for me, why is it only at this stage? And when these are missing, we don't have a direct communication channel between the customer and team. (Interviewee 3)

> And when we enter development, all these sort of demo sessions are extremely important so that we are doing it together. We are constantly reviewing if we are going to the right direction and perform corrections if needed. -- We should be able to do this more strongly together so that it won't feel like a pipe, a waterfall in a shape of a train where we put things and after a long while things come out and we see if it was what we wanted. -- how we can share and bring things in through the demos, like feedback etc. -- this is a double-edged sword as it requires activity on both sides. (Interviewee 6)

8.3 Customer value in the current development model

Regarding customer value, interviewees had diverse views to its meaning in their own perception and one used in the company. The definition presented to the interviewees by Woodruff (1997) provoked a lot of thought from the interviewees, as all of them had to take a moment to reflect it to their view and consider it to the organisation. Most interviewees agreed with it at least in its core message, but also pointed characteristics of the company to hinder its applicability. Particularly the company's domain and diverse types of customers were not easily attributable to the definition's broad message. Overall, it became clear that customer value is a widely subjective and context-dependent term, even though similar components in it can be agreed upon. A clear separation between two customer groups emerged from viewpoints of several interviewees: internal and external customers. Additionally, various forms of products and development created friction for the definition's applicability.

> Not perfect [definition]. Especially regarding what the customer is and in what sorts of situations. – The internal-external division is surely challenging to fit into this definition. (Interviewee 1)

> -- how do we define a product? -- It can be understood from several viewpoints if we develop [a product the company offers] or something else, because we also do process development and that sort of stuff where we do not develop a product: is a system people use in this [definition's] context [a product]? (Interviewee 2)

> Well yes, we use those words in a way when we talk about it [customer value], what value is delivered to the customer and how they experience this. But do we really talk that much about these things, I don't think so – I don't have a perception if people think alike of this [customer value]. (Interviewee 5)

As became clear in the challenges, the majority of interviewees felt that customer input is not sufficiently present and incorporated to the development model

currently. Many of interviewees labelled customer experience as the key component of customer value, which is where there also were existing measures in place. The majority also agreed customer value to be present in the current development model at least on a fundamental level, as development items root from the strategy of the organisation where customers are a strong focus and therefore an automatic direction for development of sort.

> -- product initiatives, guidelines in the big picture which root directly from strategy but then we also have things come up a bit different way. We have all sorts of idea channels where a lot of things surface from –- you know day-to-day operations. (Interviewee 6)

> We have funnel review and backlog grooming ongoing all the time: we constantly evaluate what we will start working on and its impact. We constantly reflect it on our confirmed strategy, available capacity, and competences. We constantly reflect on the timely customer demands and other things so we have the forums and activities in place as they should in my opinion. (Interviewee 4)

> In the current SAFe model, we aim to fulfil customer value by our best because that way we try to get the most important jobs ahead and less important ones to the end. (Interviewee 3)

Interviewees called customer value to show up in development on a per item basis due to their inherently different nature. Another interviewee on lower levels indicated that they perceived development needs to have clear backing on customer value but an overall rush to impact value delivery negatively. They also hoped for more service design activities to truly uncover value benefits.

> Well, it [customer value in development] depends on the topic, because we prioritise, and epics have varying descriptions [of customer value -related aspects]. If it is a new ordering user interface for customer service -- or if it is some sort of API or integration development, which does not really have customer value -- it depends. (Interviewee 2)

> Whatever we get from the business and marketing it just comes from the customers' demand. (Interviewee 7)

> So the impact [of customer value] would be the biggest if we really used them [service designers] in everything we do. But again, the rush and doing too many things – eats up the possibility of utilising and bringing over the customer viewpoint. (Interviewee 8)

When asked about incorporating more customer value through increased identification of development items from listening to customer feedback, interviewee described as good idea on paper, but unsustainable in practice. This is why the interviewee felt a definitive need for planning in terms of customer value as well, because improvements must be planned to ensure bigger impact with limited development resources. Even though the customer voice is valuable, they

cannot directly tell the needed development which requires coordination and planning.

> It is kind of a road of good intentions leading to hell when we try to evenly bow to everyone and be a bit of everything, which just isn't something we are able to do. (Interviewee 4)

Distinction between the nature of internal and external customers also surfaced near this topic. While involving customers in development was seen important, this was seen relevant mainly for internal customers. Service designers were utilised to assist in development targeting external customers though resources to utilise them were seen very limited.

> That we would listen to the customers, what they want and I'm sure this is done, and the views are brought over. But we cannot bring external customers to the analysis, but internal ones would be great to involve. (Interviewee 1)

> We have service designers who conduct interviews of external customers. But we cannot do this for everything and gigs for these are chosen carefully. (Interviewee 2)

As an extremely relevant viewpoint to balance customer value aspirations with reasonability from the reality of scarce resources, two interviewees also opened about the pitfalls of committing to full delivery of customer value. They described fulfilling customer value an 80% versus 20% problem, where full customer value is and should not be targeted due to the very expensive nature of receiving the last 20% of the benefits.

> We don't even always target the best customer value, that can be honestly said. Instead, we aim to optimise the cost and customer value, for sure. (Interviewee 2)

> …but the last 20% can take as much time as the first 80% purely due to the complexity increasing significantly. And then we have to do some balancing with this and when we don't have an MVP version we commit to even to the last miles and then the customer has to wait for them longer. (Interviewee 3)

The optimum amount of customer value also provoked a lot of thought from the interviewees, largely having basis on the beforementioned problem of unlocking maximum customer value being very resource intensive. Several interviewees described the company having history of developing even customer-specific customisations, which have increased complexity of the IT landscape, but on the other hand attracted additional business revenue. The need to really understand the problem instead of starting analysis with the solution in mind was seen as a key remedy to proper alignment in this regard by one interviewee. Figure 4 displays the interviewees' perception of the current state between simplicity or customer value as the main driver in development.

> Well absolutely customers are pleased if they get customisation but then again, do we view customer value through one customer or through a more generic scope of

all customers? –- does it benefit the 1% or 80% of customers and what sort of value do they get? So of course, it should be reviewed and not only through one customer. However, we have customers who bring [x] millions of euros to house in a year so of course they should be listened to -- and negotiated in a way that they [their customised solutions] would serve other customers as well. -- But they [customisations] are a major burden and difficult to get rid of. And the customers [with heavy customisations] may not benefit from enhancements done for other customers. (Interviewee 5)

Customisations have indeed turned into an enemy of customer experience and even themselves. -- The customer does not necessarily ask for complex solutions, but they ask for solutions for their problems or needs. If we understood what the customer needs, perhaps we would respond with more generic solutions which the customer would not have thought of. -- So, here's the customer-centric approach if we let the customers describe what they want instead of starting to solve it. (Interviewee 6)

Figure 4 Interviewees' grades on simplicity (0) versus customer value (5) as the target in the current development model

Overall, the interviewees expressed a vital need for activities for bringing more customer voice and therefore value to the ears of all people working around the development model. Similar actions that were listed in challenges were recognised to promote the customer viewpoint and demands. One interviewee also described visiting other offices and becoming familiar with key development stakeholders to be very fruitful to increase visibility of development.

What I suggested [to certain development stakeholders] is that sometimes when they are at home and they are not working, just go randomly to our shop and start buying things. Start buying things but maybe they don't really go to the end and purchase that item but at least try to be the end user, customer. Try to be the customer and don't know what's going on and then they will see where they can find some kind of the challenges in our shop. (Interviewee 7)

…because I have visited [other office location] and will continue to visit so that my face would be familiar because when people give us feedback and development suggestions, they feel they are not listened to, and nothing happens. (Interviewee 2)

Two interviewees proposed that they would like to see a light and agile of sort feedback loop established, which could be used to have dialogue with people

who interact with customers to confirm the direction of development and gather feedback from relevant development stakeholders continuously.

> How could we establish a sort of a light process, as we have all kinds of idea and feedback channels etc., so how could we establish a feedback loop where they [development stakeholders] could give feedback of our different functions which would be reviewed in an agile and efficient way together with the responsible persons? And they would see that ok, that one goes to the bin, that might have something, that is something else. So, do we have an agile and light process through which we could collect and purify ideas? But not so that there would be 15 people coming together and spend hours to dig the up the benefits – instead more based on gut feeling. (Interviewee 2)

> …involving the customer that you'd get insight on if it is the most important activity of the customer, in the early phase [of development/analysis]. But that validation should occur in many phases by asking if the idea is good, is the concept good, and is the design good? So, this sort of quick and agile validation should occur in the development model but in ours, at least continuously, it doesn't. (Interviewee 5)

> Right now, this feedback loop gets very slow and long when for example the lead time of an epic or feature is very long. So, the customer experience is shaped by things working slowly and problems not getting timely solutions et cetera. (Interviewee 4)

The identified challenges and connections between their root causes, effects, and effect on customer value are presented in Figure 5. The previously recognised four challenges appear as themselves with connections between other themes providing more insight on their background and effects. As no single challenge is simple on its own, the many connections between their causing factors and effects are also displayed in the figure with connecting lines. The connections and surrounding themes have been identified from the in-depth descripts of the interviewees when discussing challenges and topics around them. This figure highlights the overall difficulty of challenges of scaling agile, where issues always have many sides to them and are therefore complex.

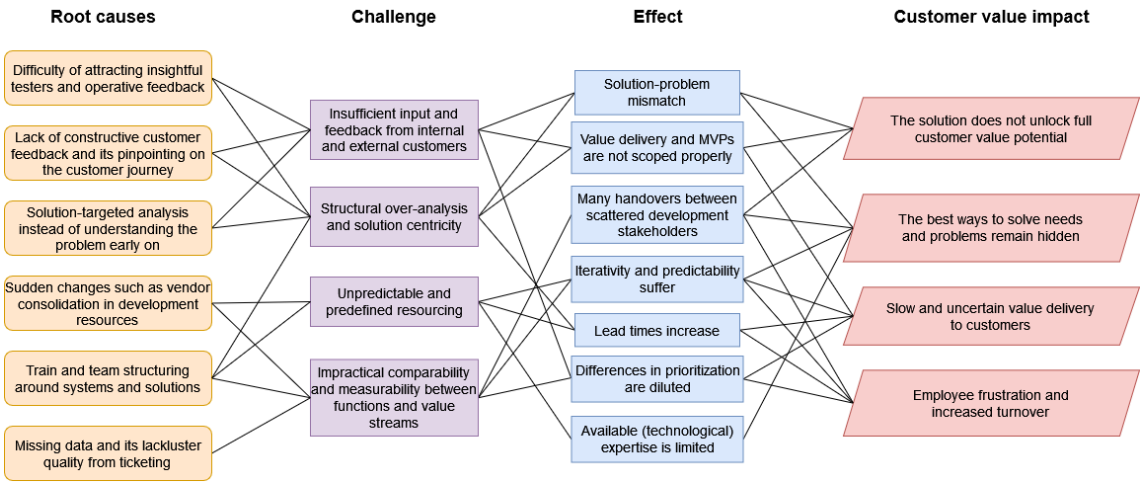


Figure 5 Root causes of the identified challenges with their impact on customer value

8.4 Measuring and monitoring customer value for prioritisation

A key requirement in incorporating customer value to the development model is to measure and monitor it effectively. The same challenge from literature became apparent in the interviews regarding customer value's measurability, when practically all interviewees described it difficult to perform representatively and accurately. Interviewees described that there are processes and systems in place to gather feedback regarding customer experience in the form of NPS and CSAT surveys from interactions with the company's services, such as online portals or contacting customer service. However, the current state of connecting this feedback to development varied per interviewee.

> [asked if customer value is measured] Well in our case no, it isn't. We don't have anything else than those hard metrics as far as I know how me, and my success are measured. And through that the success of the development function is [measured] by revenue, sales numbers, and customer experience scores. (Interviewee 5)

> I mean we have NPS, CSAT, etc. but it is very difficult to use them in argumentation because what the NPS really attributes to, we cannot directly link to the change that has been performed. So, it is very difficult to bring things visible through any sort of metric. (Interviewee 2)

> It [measuring customer value] is largely measuring customer experience: we have central KPIs, and ambitions defined for CSAT estimates regarding how pleasant the customer experiences interactions in our various touchpoints. -- So, if customer value is perceived as a great customer experience – if a great customer experience produces value to the customer – then it is really present in what we do. (Interviewee 4)

The core issue of attributing customer value in numbers was prevalent in several interviews. In general, interviewees did not feel that customer value itself was measured. An interviewee presented an idea that perhaps an indicator reflecting how much a customer has centralised their services to the company's offerings could indicate this. This could also be expanded to grasp aspirations and future directions to increase understanding of customers. Interviewees generally pondered around whether metrics actually measure customer value, or internal business value.

> If I understand currently, we should get more centralised and better data about that [customer value]. But I would say the biggest problem to be converting it to euros. Or let's say to a tool in prioritisation. (Interviewee 6)

> Perhaps if we could estimate value per customer on how much they centralise their services to us etc. And how satisfied they are, indication from measuring satisfaction that where the customer could be headed, should we focus more on them and what services they use and what we should therefore improve. But this is difficult. (Interviewee 5)

Measuring and tracking of customer value was seen to be centred around specific roles, whose responsibility is to inspect how development has impacted things after it has been implemented. They had been quite recently implemented were still in the process of maturation.

> Well yes, we have these business benefit realisation managers [in certain segments] who oversee what sort of benefits are generated when we push things to production. (Interviewee 1)

Overall, regarding prioritisation, interviewees closer or on team level focused strongly on measuring development work itself and improving its measurability and predictability. Customer value was not currently among these but was seen as a potential and beneficial addition. They felt that the themes they picked to development had already been prioritised according to business value on higher levels.

> Yes so, we have common level measurements such as feature lead time, PI accuracy, feature throughput -- Well yes – [in terms of] how our prioritisation works and how customer voice is heard there – business strongly prioritises our doings starting from the epic. They set certain priorities for the epics which are inherited for capabilities and again for features. So teams don't choose the features based on what they would like to do: of course there's some sort of business need. (Interviewee 1)

> If we started to report our work more based on the customer benefit rather than how many tickets or features have been completed – I feel it would be it [how to measure and monitor customer value better] because you get what you measure, you know? -- If we could somehow implement it [customer impact] to our metrics, then its realisation could be inspected among other metrics afterwards. (Interviewee 6)

Similarly to customer value's general role in the development model, when asked if prioritisation was currently based on customer value, interviewees mostly agreed. A strong ambition to focus prioritisation on customer value was apparent from the answers. Still, other factors, mainly business revenue, were seen to surpass customer value in prioritisation of development items.

> Yes it [prioritisation] should be [currently] based on that [customer value] because there are certain business lines who under their own themes prioritise certain things which then start to get worked on. They should surely indicate what the customer needs, be it internal or external: where we get business benefits and value from. (Interviewee 1)

> -- everyone knows that in business you prioritise what creates the most value. But what is the value of customer value compared to raw euros? (Interviewee 6)

Two interviewees did not see revenue and customer value to exclude one another, because they are interlinked. Customer value through customer experience, for example, is an important differentiator and potential competitive advantage particularly in the telco industry. Customer value was therefore seen as a subcomponent of revenue of sort. The interviewees' grades among various

levels regarding how the development currently incorporates and prioritises on customer value are visible in Figure 6.

> Like for example [when] business management wants to prioritise things, it is revenue which is the number one and customer experience after it. -- if the service provides value for the customer, it does provide revenue to us because customers want to buy it more and more. (Interviewee 5)

> If we think about customer value – if we think about the telco business. Some wise people have said it well that in the end it is customer experience which differentiates us [service providers]. Of course, we have different products – but you know it is customer experience through which we can differentiate ourselves by being a better operator and then it means that it is a reason to choose to be our customer. And that is revenue, so I don't think they contradict each other at all. (Interviewee 6)

Figure 6 Interviewees' gradings from 0 (minimum) to 5 (maximum) how the current development model incorporates customer value in it and its prioritisation

In terms of actual prioritisation, the interviewees considered that it would be greatly beneficial to incorporate customer value to WSJF prioritisation formula and prioritisation model in general to prevent too barebones MVP-implementations from entering the market and creating suboptimal compromises with available resources.

> …when we try to replace services to which we have put years of effort with MVP-based thinking, they [results] aren't very good in terms of customer experience. And that it reflected in numbers regarding customer feedback and even sales sometimes. – it [WSJF] drives us to MVP thinking, where we produce maybe lacking solutions. And when a customer experience metric isn't included in it [WSJF], it is a major risk as customer experience often takes a dive when we create an MVP solution and use it to replace an old solution, for example. (Interviewee 5)

> [we should have] very flat discussion when prioritising so that we'd really understand which option creates what sort of work and what the outcome might be. Because ultimately, it always is some sort of a compromise. (Interviewee 8)

The prioritisation process was described to consist of multiple phases, where items are analysed, and the most critical ones identified from the candidates. WSJF was merely used as an assisting tool, not as the ultimate formula to base decisions on. Another interviewee also perceived current prioritisation as lack-

lustre, because vastly different development items were compared by their "face value" not their true nature. Also, one interviewee explained data quality challenges to render WSJF impossible to utilise at the moment.

> WSJF value treats all things and tickets very equally, but this is a thing we are missing. We do not guide tickets based on their WSJF value but instead on best understanding which ticket is more important than other. And this is sort of another of our repeating sins troubling our technical development. (Interviewee 3)

> We try to see how much we can generate revenue with developing this or improving customer experience of our customers and so increase customer retention. -- So those are perhaps the most important criteria which kinda force us to prioritise. -- We have WSJF estimation alongside where we estimate cost-benefit ratio how much it will take time and money and how much it will provide money or other benefits. (Interviewee 5)

> I'd like to [use WSJF]. Yea I'd like to utilise it here, but our biggest problem is that we don't have the data. We don't have proper business cases and we don't have estimates of the amount of work. (Interviewee 3)

Basing all decisions on a singular formula should also consider the special characteristics of the company's domain. Interviewees added that there are always subjects that can pass all other development items, such as regulatory, compliance and security requirements which eat space from other items in the development queue. The heavy regulation of the telecommunications industry highlighted the regular occurrence of such items according to the interviewees.

> Regulatory requirements cause us a lot of work which take priority over the development business would want. So, if we say that we have a change regarding GDPR, for example which must be done, it will be done before developing [certain product] or other equivalent things. They always cut ahead of these things and steal development resources. (Interviewee 3)

> Well, some say that it [the current model] is inflexible when a program increment lasts three months and they feel that they have to wait the three months if [their] needs or requirements did not progress to the implementation of trains. Of course, it is not unheard of that some regulative, really highly prioritised things come up which must be done. Then we do replanning. (Interviewee 1)

9 DISCUSSION

This section reflects the earlier findings and the literature review to answer the research questions of this thesis. The openness of the interviewees and the successful formulation of the interview questions reveal relevant remarks about the current state of the development model and customer value. There are many links between the interview themes and the literature, which are discussed alongside other notions of interest and relevance. The interviews also moved the buzz of SAFe's own material aside as interviewees presented practical and balanced views of the framework and its adaptation, providing insight for scaled agile research and customer value's realisation in practice. The most essential points of interest are highlighted with listings and/or cursive styling to underline their importance towards the contributions of this thesis.

9.1 Discussion of challenges

The research questions of thesis were laid out to two parts. The first RQ1 and 1.1 were targeted to explore challenges of scaling agile and RQ2, RQ2.1, and RQ2.2 towards the role of customer value in (scaled) agility. First, literature of challenges was explored with the following RQ1:

RQ1: What challenges are there in scaling agile and its methods?

In total, five types of challenges were recognized from existing literature. The types were conceptualised based on their common aspects. These types with their respective references were first presented in Table 2, and are the following:

- Methodological challenges
- Conflicts with other methods
- Cultural challenges
- Domain-specific challenges
- Complexity challenges

To gain contributions from a practical context, sub questions were created to reflect the findings from literature to a context of practice. RQ1.1 was aimed to reflect the presence of earlier findings on the challenges of the target company's scaled model. This similar format continues in the customer value related research questions. RQ1.1 of this thesis is goes followingly:

> **RQ1.1:** Do the challenges in the target organisation correspond to ones found from literature?

To answer it, the challenges of the target company's current model are mostly in line with those identified from the literature. There are some differences which is natural as the literature discusses the topic in a broader context, whereas the interviews of this thesis uncover a specific context with its own history and characteristics. Many of the challenges faced by the target company fall into the following categories of existing research presented earlier:

- Domain-specific challenges
- Complexity challenges
- Methodological challenges

The other two categories of challenges from Table 2 were not nearly as apparent as the previous three. *Conflicts with other methods* were not observed, as the company does not have a notable history of pursuing other (scaled agile) methods other than SAFe. *Cultural challenges*, mainly difficulties pointing to early stages of agile transformation were also not apparent, as the development model has matured and seen adaptation since its inception. The issue of alignment between autonomy and planning was also not considered problematic by the interviewees and therefore cannot be listed as a challenge, despite its presence under cultural challenges.

However, one aspect of *cultural challenges*, the almost classic problem of doing agile rather than actually being agile (Ranganath, 2011) did emerge. One interviewee also mentioned that although ticketing tools can be designed to work with almost any development method, some people in the organisation assimilate a ticketing tool and SAFe as synonymous. Lacking knowledge about agility and methods was also a prominent theme in the literature with Edison et al., (2022), for example, mentioning the pitfall of attributing agility to a tool instead of a radical and fundamental transformation.

Domain-specific challenges surfaced regularly in the interviews. This is understandable, as they are particularly applicable to the highly regulated and competitive telecommunications industry. Several interviewees recognised that new regulatory and security requirements take valuable development space from other items, because by their mandatory nature, they cannot be neglected. In practice this meant that they received special treatment in prioritisation. The nature of security requirements and priority of their needs in development was highlighted by Fitzgerald et al., (2013).

Complexity challenges were another prominent category of challenges present in both the literature and the target organisation. Large organisations are often complex, which create additional obstacles to be solved. Hobbs & Petit, (2017) and Eklund et al., (2014) emphasise the need for careful planning to solve them accordingly. This was also accurately described to affect the complex development of solutions in the company. This issue receives more attention in the next subchapter where customer value is discussed.

Interviewees provided many details that allow deeper exploration of the *complexity* and *domain specific challenges* through the lens of SAFe. Organising around value streams, one of the key concepts SAFe highlights (Knaster & Leffingwell, 2020, pp. 135–137), was one of the clearest challenges in the target company. By their definition, value streams in SAFe are a guiding structural component, that should steer the building of solutions to a more comprehensive scope by understanding the value solutions create around them rather than locking the perspective on a single solution (Knaster & Leffingwell, 2020, p. 83). Currently, the target organisation was indeed organising development around segments and systems and was therefore not fulfilling SAFe in this regard.

The interviewees described many symptoms which can be traced to this issue. Firstly, treating development teams and trains as equivalent in terms of measurability was perceived as problematic as they were too different to be measured and compared using the same means. Also, limiting resource acquisition to meet the needs of individual solutions reduced the (technical) competence of developers, and created organisational silos where flexible resourcing and reduced number of handovers could not be achieved. Many handovers were seen as risky and often resulted in the next expert having insufficient information available. Increased handovers in context the of complexity of large organisations have been identified as major painpoint in literature by Šmite et al., (2017). Figalist et al., (2019) added that when development environments cross organisational boundaries, the many steps in communication chains create a similar problem. This is very applicable to the target company due to its high use of external suppliers and consultants.

Overall, as intended and emphasised by SAFe, the target organisation has created customisations to the development model that have been seen as relevant and impactful for the better. Many interviewees called Finland as a leader in SAFe implementation compared to other countries in which the company operates. This forerunner position has also caused some friction when collaborating with other countries or common functions that are still more or less finding out the best way to do things.

This issue of varying levels in internal agility resembles the *methodical challenges* which were identified from literature. Other country units that lag behind in the agile transformation have not addressed the adaptations of the leading implementer but have instead pushed Finland's more mature model towards a more vanilla state in the name of common ways of working. This is a difficult point to find a balance on, as country units have their own characteristics and therefore require unique adaptations and time to implement them. Different

levels of agility between internal functions have been identified as a challenge in the scaled agile in literature by Kalenda et al., (2018).

Despite the focus on challenges, it was welcome to see that the current development model was still seen in an overall positive light. Its structure, cadences and formalised way of working were seen by interviewees as providing useful benefits and overall, a clear enhancement on previous ways of working. The openness to criticism and ambition to iterate the model further were also encouraging signs which were even mentioned directly in the interviews. SAFe itself sets learning from previous iterations as one of its key practices in the form of continuous improvement which is also essential is agility as a whole (Knaster & Leffingwell, 2020, p. 131). In this regard the agile transformation of the organisation, probably the biggest make-it-or-break-it -moment of the SAFe implementation, can be considered successful, as relevant adaptations have been made and new improvements are generally welcomed. The latter also speaks for successful fostering of an agile culture, which SAFe identifies as a requirement of lean-agile leadership for the spread of agility in organisations (Knaster & Leffingwell, 2020, pp. 111–112).

Although changes to improve the development model were welcomed, the interviewees also explained how constant change initiatives and restructures inflict stress and other negative effects on the staff. Kalenda et al., (2018) describe that restructuring the organisation to support agility poses serious challenges as moving resources from one place to another can create mismatches in their supply and demand. This had certainly realised in the organisation, when restructures, despite of their good intentions and potentially positive outcomes, created periods of reduced efficiency as people and their processes were still finding their places. This resembles Kalenda et al.'s (2018) description of quality troubles that emerge in the process of agile transformation and shift in balance between autonomy and planning.

On a more positive note, it was great to witness the commitment that the current SAFe-based development model enjoys from the very top of the company. This is also a critical success factor for enabling agility especially in its adoption by the organisation (Knaster & Leffingwell, 2020, pp. 111–112). This commitment and culture of continuous improvement may very well be some of the factors behind the successful agile transformation of the company. One may wonder how ways of working can even be improved at all if there is no clear method or support for doing so. New initiatives focusing on benefit realisation and pre-analysis of development items were also underway with no sign of trials stopping. Many interviewees pointed out that while the current model was certainly not perfect, they still could not see other way the development could be arranged better.

Lastly in terms of challenges and moving closer to customer value, the lack of customer input was identified as a major concern by the interviewees, who emphasised the importance of understanding customer needs in the preliminary stages of analysis before technical development. This was not a major theme in the scaled agile literature, although customers are one of the most im-

portant aspects of the Agile Manifesto (Beck et al., 2001), which can be considered the philosophical foundation of agile methods. This is another aspect where the strong customer-centric focus of target company and its domain is evident. Interviewees identified customer input as a current challenge and would welcome concrete measures to increase its formal say in prioritisation and the development model as a whole.

The particular interest in customer input is also indicative of the maturity of the current model, as the focus shifts from transformation challenges to steering the model towards gaining competitive advantage based on the characteristics of the company's business domain. This makes the challenge of lacking customer input and its incorporation into the development model to resemble *domain-specific challenges*, under which it can be classified. Telcos in general have a particular interest in differentiating themselves from the competition through customer experience (Penkert et al., 2019). Perhaps this is why customer value is not as prevalent in scaled agile literature, as its role is not as dominant in other industries even though customers are at the core of the agile manifesto itself. In this sense we can consider the target company to be on the forefront of exploring the tight incorporation of customer value into its development model.

9.2 Discussion of customer value

The second part of the research questions were crafted with strong focus on customer value to research how it could be used as an objective tool in development prioritisation and decision making. An established research stream of customer value was found in the literature, but the lack of consistent definitions and methods quickly became apparent which was also noted by Zeithaml et al., (2020) and Sambinelli & Borges (2019). This shifted the focus of the answers closer to the empirical findings, although additional insights were gained from the research of customer value's subcomponent, customer experience, and its measurement with NPS. In addition, existing development prioritisation methods and metrics were explored from a customer value perspective.

To study the practicalities of customer value, it was first necessary to clarify its definition. A common definition of customer value is necessary to align views, which has been found particularly difficult in scaled development environments (Kasauli et al., 2017). As a term, customer value does not appear as unanimous in the literature, which was also the case among the interviewees, all of whom took some time to apply the definition by Woodruff (1997) to their own perceptions. Many questioned its applicability to more specific contexts, such as the internal/external nature of the target firm's customers but still mostly agreed on its general meaning. This is consistent with the multifaceted nature of customer value in academia, where different schools of thought have emerged because the term can be discussed from many angles (Zeithaml et al., 2020).

When reflecting the interviewees' views on Zeithaml et al.'s (2020) five elements (who, how, what, where and when, why) of customer value research paradigms, many points emerge. Answers pointed to the perceptual nature of customer value's accumulation, as the overall definition that the interviewees often had was oriented towards customer experience which is a strong strategical focus of the company. The company's domain in general also shares this interest (Penkert et al., 2019). However, when discussing customised development, there was a strong individual element to the creation of customer value. Perhaps this is common in B2B environments, where the impact of the individual customer may be greater, and the customer may therefore expect more.

The interviewees touched on the aspect of '*how*' by wondering how customer value could be captured and more accurately measured. A strong focus on perceived and experienced nature was evident as there was no clear answer and service design activities were regularly practised to accurately understand customer's value ambitions. In terms of the '*what*', the interviewees' strong attribution of customer experience as the main component of customer value's dimensionality was apparent. No other factors emerged or were noteworthy for the importance of experience.

The questions of '*where*' and '*when*' did not arise in the interviews. This was to be expected as value was discussed on a general level with no focus on individual offerings and their specific customers. However, interviewees did mention the benefits and challenges of accurately capturing feedback from specific parts of the customer journey when discussing other issues. This was seen as providing insight into where potential customer value pitfalls might lie. In the final question of '*why*' a link from the interviewees' answers can again be drawn to customer experience, which was undoubtedly the main area of interest. The interviewees attributed great customer experience to increased revenue and an overall differentiator in the industry, which have also been recognised in the literature by Reicheld & Markey (2011) and Penkert et al. (2019). Moving on from definitory issues to methods, the second area of interest in this thesis, customer value, was first was explored with the following research question:

> **RQ2:** What models and methods are there to measure customer value and use it as means of prioritising agile development?

As mentioned in the beginning of this subchapter, proven methods to measure customer value in the interest of RQ2 were found to be scarce, which was also supported by Sambinelli & Borges (2019). The found methods, such as story or benefit points, rely on a largely sentimental and gut-feeling -based approach to identifying the customer value of development items and assigning a score to them. This can be seen as undesirable, as subjective views behind prioritisation reduce the transparency of development decisions and can therefore have a negative impact on the development environment if people do not agree with or feel involved in these decisions (Cleland-Huang, 2015). Power (2011) also highlights the nature of organisational silos and their tendency to promote individual efficiency and value creation, which hinder shared value ambitions.

The lack of proven methods to quantify and measure customer value shifted the interest towards customer experience, which does not suffer from these issues at least in similar extent. Customer experience, its measurability with NPS, and positive effect on business revenue (Reichheld & Markey, 2011) have attracted research interest and can be seen as an enabler in measuring customer value. This is because customer experience is one of the subcomponents of customer value (Zeithaml et al., 2020), which itself lacks measurement methods (Sambinelli & Borges, 2019). Studying the current status of prioritisation on customer value in the target organisation was the set as the goal of research question 2.1, which was the following:

> **RQ2.1:** How does the target organisation promote the role of customer value in development prioritisation?

To answer RQ2.1, customer value was certainly present in the company's current development model but not in a formulated or other established measure. Currently, measures to incorporate the voice of the customer into development were mainly in place at team level, where teams were regularly fed feedback from the systems they were working on. According to the interviews, such processes were not in place at higher levels and early analysis was perceived to suffer as a result. This was also the case for prioritisation, which was heavily based on perceived business revenue.

Interviewees considered that prioritisation decisions were heavily concentrated around specific roles and even gut feeling. Involving different stakeholders in prioritisation decisions was still practiced which alleviated some of these concerns. Many interviewees deemed basing prioritisation on a calculation formula as the optimal method to avoid the issues described in literature. This was the case although the company had implemented ways to lessen their severity through more transparent discussions. Even though business revenue was seen as linked to customer value, interviewees still felt that customer value was not adequately considered, mainly due to the lack of quantification methods, which made it difficult to base decisions on.

Despite the lack of sufficient customer value incorporation on higher levels, customer value was perceived to have an indirect presence through customer experience metrics (NPS, CSAT score) and their perceived link to business revenue. Despite the fact that customers are at the core of the agile manifesto (Beck et al., 2001) and are on the forefront of the company's strategy, interviewees considered business needs, usually revenue, to ultimately surpass it in development decisions. These challenging realities of measuring customer value also unfolded in answering the following research question 2.2:

> **RQ2.2:** What means are there to increase this (the role of customer value in development prioritisation)?

To answer RQ2.2, we must consider the characteristics of the company's business domain and the configuration of the current model to understand its limitations and realities. Several interviewees considered WSJF to be the optimal

prioritisation method as a formula, but it was not currently used for the company's development activities. This was due to infeasibilities in its vanilla configuration for the company's development requirements, such as development of security-intensive items and other time-critical development, such as fulfilling regulatory requirements.

The previously discussed interest and existing customer experience measurement practices promote its usability as a factor in prioritisation. This is particularly relevant given the telco domain's interest in customer experience. SAFe itself suggests basing prioritisation on the WSJF formula (Knaster & Leffingwell, 2020, p. 154), which does not include a direct customer value metric. One of its components, Cost of Delay can be considered as value related, as if development work is delayed, its impact remains dormant. Combined with the interviewees' views on the link between business revenue and customer value, CoD becomes a potential factor to expand upon, especially together with customer experience scores. This could pave the way for a prioritisation formula where customer value through customer experience scores influences CoD and is therefore included in a more objective manner. This means that *WSJF requires adaptation for it to be leveraged effectively as an objective, domain-, and customer value-inclusive method.*

Expanding WSJF with a focus on CoD should also address the challenges discussed in the previous subsection regarding the company's domain, which has implications for the prioritisation and development overall. Interviewees described certain types of development, mainly security or regulatory requirements to always surpass others. Similar themes have been identified in the literature, including security requirements that cannot be built piece by piece without complete plans (Fitzgerald et al., 2013). The nature of certain products may also prevent the full use of iterative and incremental development as they require guiding structures to base development on (Eklund et al., 2014).

Another interesting angle that emerged from both literature and the interviews was finding the optimal balance between customer value and the complexity that delivering full customer value can add to offerings and the IT environment. Interviewees agreed that while customisation provides customer value and therefore business revenue, its tendency to increase the complexity of the company's IT environment should be avoided. Transforming the last remaining pieces was seen extremely expensive both in the interviews and literature by Conboy & Carroll (2019) and often not even mandatory to unlock the full value potential. Several interviewees described this 80-versus-20 problem as often occurring in development, where the first 80% is delivered as planned, but the last 20% proves to be very resource intensive to deliver. The costly issue of fully meeting customer expectations has been addressed in research by Ashmore & Wedlake (2016). Overall, this issue highlights the importance of *proper scoping to target development resources to unleash the most value with finite resources.*

Overdelivering on customer value aspects can have many side effects. These include clogging the development funnel if the prioritisation and scoping of development work is not properly managed and too many items are worked on at once (Moreira, 2017, p. 139). Defining the optimal scope of MVPs and de-

velopment items in general was perceived as difficult by the interviewees, as too barebones solutions reduce customer satisfaction if they are used to replace deprecated solutions that have seen customised development over the years. The interviewees also highlighted the need to *limit the size of development items to prevent single items from consuming all resources*. The company could benefit from an focusing on the viewpoint of MVPs described by Ries (2011), where instead of just treating it as the stripped-down, barebones first implementation, its ultimate purpose should be to serve as a platform to gather as much feedback as possible to steer development to the direction of the most customer value.

SAFe, as agility in general, promotes continuous improvement, which is rooted in the iterative nature of agile development, which aims to constantly learn and build better solutions based on this feedback loop (Knaster & Leffingwell, 2020, p. 131). The prioritisation of development should also behave in a similar way to *continuously assess whether development is aligned with customer value*, also proposed by Olsson & Bosch (2015). This would prevent situations where the last miles of value delivery prove unexpectedly costly. Some interviewees considered the very long nature of these feedback loops to currently hinder efficient and effective value delivery.

A light feedback cycle between operative staff familiar with daily lives of the customers, and development analysts familiar with development and IT realities was suggested as a way of continuously providing valuable customer insights to development to ensure that the right things are done in the right way at the right times. In agile methods such as Scrum, so-called retrospective sessions are held after each development sprint to gain feedback to improve development (Schwaber, 2004, p. 20). A similar cycle in the company to connect the previous two realities would be highly valuable.

Iterativity, incrementality, and best decisions through team autonomy are fundamental concepts which form the benefits of agility (Beck et al., 2001; Larman & Basili, 2003). However, large organisations use planning as a means of managing their inherent complexity which contradicts with the freedom and trust experts have in agile way of working (Moe et al., 2021). The interviews showed that this balance leans strongly towards planning in the target company's model. It also seemed that the current model had been particularly tweaked to managing resources efficiently on the lower levels near technical implementation, which teams still could influence.

One might ask whether development is really agile at all, given that teams are handed pre-prioritised needs and requirements in a very specified form. This naturally tends to lock in perspectives at other levels as well. Then again, teams were perceived as having technical autonomy and development work was reviewed and iterated accordingly at the team level. The customer-orientation of the current model can also be questioned as the fundamental benefit of agile teams, diverse experts making the best decisions with a strong customer focus, is clearly not being fully realised. Incorporation of customer feedback at team level was considered to be at a satisfactory level, the lack of which at higher levels was a concern.

Despite these issues, the interviewees largely did not regard full agility as the ultimate goal or remedy for current issues. The interviewees' general satisfaction with the current alignment supports the tweaked state where development work is generally quite precisely specified, for which the according development resources are contracted for. On the one hand this means that development is carried out with a top-down approach, where needs and priorities are mandated from high levels to lower ones. One the other hand, there is still some agility, as team members have a say in technical matters and higher levels respect constraints or changes that may occur in development.

Developers and higher-level experts were also not close to one another which further widens the gap between them. The current autonomy-planning balance and lacking organisation around value streams do not encourage initiative from developers which one might consider fully agile development to require. For future needs the company should *review this balance to match competitive needs*, a task which Moe et al., (2021) describe to be very dependent on the characteristics of the organisation such as culture, the model's implementation's maturity, and the domain.

When considered with the large and complex IT-environment of the company, the current balance is consistent with the typical use of planning to manage complexity rather than allowing team autonomy in these kinds of environments. One interviewee framed the issue nearly identically as Moe et al., (2021), describing full team autonomy as impossible in a large development environment. Still, this balance described as delicate in the literature by Conboy & Carroll (2019) and Moe et al., (2021) did not seem as such in the company. This was possibly because the team level had seen the most tweaking and regular use of external development resources. Nevertheless, one can wonder whether the value delivery which has been seen to occur through resource integration in B2B environments (Macdonald et al., 2016), is at the best possible level in terms how development resources are currently acquired and organised internally. If development were to shift towards greater autonomy, more integrated resources might be required to do it successfully. In practice, this could mean *more in-house resources or at least longer supplier contracts* which would increase commitment with perhaps even *incentivise service providers* to increase customer value in the target organisation through their work. Further practical and academic implications and contributions are discussed in the following, concluding chapter of this thesis.

10 CONCLUSIONS

This section sums up the essential aspirations, findings, and contributions of this study. The contributions and their implications for research and the target company are described, followed by the limitations that have been recognised. This thesis has managed to bridge two inherently different but fundamentally related areas of scaling agility and customer value under a relevant scope. By identifying the links and gaps between them, many academic and practical contributions have emerged. These can be used as a reference for companies in their agile endeavours on how to avoid certain challenges and finding ways to promote the role of customer value in development activities. This thesis contributes to scaled agile research by providing information from a more mature context of practice alongside a customer value lens. This chapter is the last of this thesis, followed only by the list of references and appendices.

The main objectives of this thesis were to research scaling agile, its challenges, and the role of customer value to find ways to base development decisions, such as prioritisation on it. Firstly, a literature review of the fundamentals of agile was conducted through its evolution from original team-level agile methods such as Scrum to scaled methods. This study has focused on SAFe, which is the most popular scaled agile method today (*16th State of Agile Report*, 2022). Scaling agile has seen increasing interest in practice due to its benefits for large organisations in today's intense markets, where born-agile competitors require appropriate responses (Ebert & Paasivaara, 2017; Uludağ et al., 2021).

From the history of agility and the birth of scaled methods, the topic moved on to map their challenges. Scaling agility from its original contexts to larger ones has its unique obstacles (Kalenda et al., 2018). The challenges that were identified from existing research are methodical challenges, conflicts with other methods, cultural challenges, domain-specific challenges, and complexity challenges. Research of these challenge types was grouped in Table 2.

The final part of the literature review sought to shed light on customer value, its quantification, and how it could be used as basis for decisions in (scaled) agile development. Companies aim to be customer-centric to provide the maximum value to their customers to attract business (Conboy & Carroll,

2019; Ebert & Paasivaara, 2017). Despite customers being a central focus of agile development philosophy (Beck et al., 2001), companies such as the target of this thesis still struggle to realise customer value in an impactful and measurable way. Direct and established methods for quantifying customer value were lacking in the research, which Sambinelli & Borges (2019) had also found. Other methods for objectively incorporating customer value into development were also scarce. This highlighted the challenge of studying value, which can be described as an abstract, largely sentimental, and subjective concept. The lack of an agreed upon meaning for it in research (Gallarza et al., 2011) also supports these defining characteristics and the uncertainty of its precise definition, which was also noted in the interviews. This uncertainty around its definition also hinders the development of new methods and their generalisability to other contexts.

To reflect the findings of the literature review into practice, employees at various levels of SAFe in the commissioning company were interviewed using a semi-structured interview method. The interview data was analysed using thematic analysis to identify common topics between them. The challenges from the literature were found to generally apply to the target company as well, although the maturity of its scaled model had the shaped current challenges closer to the characteristics of its domain and operations as a whole. This means that as the agile transformation progresses, the challenges faced by the organisation may also change. The same applies to the development model and the organisation's needs for it and in the case of the target company, the model had been adapted to suit its own needs.

The subcomponent of customer value, customer experience (Zeithaml et al., 2020) has attracted research on its link to revenue and measurement with NPS (Valentina et al., 2018). In highly competitive markets, such as telecommunications where the target company operates, customer experience is regarded as a means of differentiation (Penkert et al., 2019). This also surfaced in the interviews, highlighting its role as a factor to base development decisions on. The lack of other measures that reflect customer value points towards utilisation of NPS on how customer value's role could be promoted. Currently NPS was not present as direct or objective component in the company's prioritisation decisions. Interviewees considered customer value to influence decisions indirectly through its link to business revenue, which was considered the dominant factor behind decision making. There was high interest in basing decisions on an objective prioritisation mechanism such as a calculation formula, which was also promotion in the literature by Cleland-Huang (2015). This sparked interest in adapting SAFe's default prioritisation method, WSJF, to the company's domain-specific needs mainly in terms of customer experience and security.

Despite the recognised challenges, the interviewees agreed that the scaled development model has improved operations when compared to the previous way of working. In this sense, the adaptations of the model and the agile transformation of the company to this point can be considered successful. This is not an easy task. While the model is not perfect, the very nature of agile methods is

to continuously improve, which the company has done throughout its agile journey. Nevertheless, the general success of the adoption and adaptation of SAFe 5 spawn optimism for future improvements. This once again demonstrates the vitality of an agility-embracing organisational culture for the radical nature of agile transformation (Knaster & Leffingwell, 2020, p. 107; Misra et al., 2010). The contributions of this thesis for the target company and practice are presented next, followed by academic contributions and limitations.

10.1 Contributions for the target company and practice

This subchapter presents potential remedies the target company could consider adopting to mitigate the challenges identified in the findings. It also presents actions to increase customer value's role in the development for the company to trial. These ideas are largely based on the interview material with supporting statements from the literature where relevant links were found. They also provide generalisable contributions and references for other organisations that have progressed from initial stages of agile transformation to a more mature state, particularly in customer experience-oriented markets. The suggested actions for the target company to consider are the following:

- Establishing a feedback loop between operational and development stakeholders
- Setting monetary and learning incentives to motivate testing participation
- More value-stream aligned development structure
- Introducing customer's daily lives more regularly to development stakeholders
- Contracting services with a customer value/experience incentive
- Basing prioritisation on adapted WSJF with customer value and domain-specific characteristics accounted for in Cost of Delay
- Defining the adaptation level of individual country units and common Way of Working to enable successful adaptation and common guidelines
- Assessing autonomy versus planning balance continuously to meet market demands

A major pain point in the current model was the amount and incorporation of customer's voice to thoroughly understand needs and limitations. This issue was closely related to the over-analysis and solution centricity of the current model, as shown in Figure 4. Ideally, solutions would be crafted on a customer problem-based approach. After understanding the customer need rather than just the demand, the according solution would begin to take shape around technical and other capabilities.

The remedies for these two issues are twofold: firstly, *customer views should be more integrated into earlier phases of development*. In the interviews, it appeared that at the lower levels of SAFe, the voice of the customer was regularly included to improve the developers' understanding of the requirements. However, this did not seem to be as well established at the higher levels where the initial phases of analysis and prioritisation took place. Secondly, *treating MVPs as the basis for the earlier feedback loop* similar to Cleland-Huang's (2015) approach rather than the barebones implementation of new solutions, development would naturally be more customer-driven.

A practical suggestion for higher levels would be to create *a light feedback loop* between operative staff and development analysts and other stakeholders where they could get answers quickly to small but impactful issues without having to take extensive action. For minor issues, a chat group could be a practical way to achieve this. For larger issues, meetings of this established network of operative staff and experts could be the way to gain insight into larger issues. Increased customer feedback would also help mitigate current troubles in scoping and solution-locked analysis.

However, these practices are useless if operative staff are not motivated to participate in testing or do not see its benefits. To overcome this challenge of getting input in the first place, *the company must provide incentives to motivate participation to testing activities*. As operative staff's salary models are often based on sales commissions, these incentives must compensate for the loss of income that testing time generates. Adding a development value creation component to their existing sales commission models would make it easier to quantify the time spent on assisting development, for example. The size of incentive could be static or even based on the CoD or business case of the development item.

The introduction of financial incentives naturally means additional development costs. However, it is important to remember that it is extremely costly to make changes to development which has been misdirected in the first place due to a lack of feedback, not to mention the additional benefits the early feedback could provide. Insightful feedback can even reduce the workload or unlock much more value from the development than was initially thought as problems and their relation to the day-to-day activities are understood. Alongside financial incentives, the company could consider treating the time that operative staff spend on development as a source of learning. Eager employees may be voluntarily willing to pitch in their feedback, especially if the opportunity is regarded as a learning one and therefore beneficial to themselves too.

Overall, *motivation should be considered as an investment to gain more value from development*. Given the company's regular reliance on external consultants, the nature of their service contracts can also be considered. Contracting services with an additional incentive based on customer feedback has been found to be beneficial (Macdonald et al., 2016), which the target company could also implement to its contracting model. This could also be used as an inspiration when considering the incentive model for attracting testers and participation in development.

Alongside the motivation aspects, *the daily tasks and "life" of customers could be brought up more regularly at the higher levels* to familiarise experts with their needs and interests. Listening to customer service calls, visiting stores, reading customer feedback, or even participating in operative work could provide immense indirect insight for development experts to keep their perspective aligned with customer value. This could also ease the current solution-centricity to shift the mindset from technical and other capabilities to the customer side. A quick example action would be to direct existing customer service newsletters also to development stakeholders. These reports are written by operative staff and include comments on the types of issues have surfaced in customer interactions during the day.

Another major issue in the current development model was its structure, which was not aligned with the company's value streams. Organising around streams of value aims to structure the development of smaller items with a common value aspect under portfolios to ensure synchronisation and collaboration to deliver the maximum customer value (Knaster & Leffingwell, 2020, p. 104). For the company, this would mean *reforming the current segmental structure to correspond value streams* to unlock additional benefits of agility.

The ultimate, truly agile way of organising around value streams would be to structure both operative staff and development experts into the same teams, where views could be exchanged effortlessly without heavy handovers. This would also alleviate the lack of feedback by shortening the much-needed feedback loop between development and operative staff to realise within a team. As development would learn and benefit from the operative expertise, this would also be a great learning opportunity for operative staff to learn new skills and aspects. The recent rapid rise of generative artificial intelligence and its application to customer service highlights the necessity this learning could provide for them. However, as with any change, decision makers should consider the concerns raised by the interviewees about the pressure that constant change places on staff and the negative impact this has on their work.

In addition to the benefits of synchronisation and bringing relevant development actors closer together, greater value stream alignment could also ease the challenges of measurement and comparability. Examining similar development activities within a common stream would increase their comparability. Also, the balance between the line organisation and SAFe structure is an additional point to consider, as slipping roles from one to the other can cause confusion. Perhaps the line organisation should aim to break organisational silos to support the virtually organised value streams, which are realised in the line organisation in the form of trains and their agile teams.

The challenges of measurability were perceived to affect the current prioritisation model, which was not based on an objective method such as WSJF. Interviewees found WSJF to be a useful but lacking tool in its standard configuration, because it did not suit the company's domain-specific needs. Many interviewees indicated potential in basing prioritisation decisions on a formula which would reflect customer value through a customer experience metric.

Customer experience's status as customer value's subcomponent (Zeithaml et al., 2020) and prevalent measurement of NPS in the company suit its integration to WSJF.

As presented in the discussion, *a new prioritisation formula based on WSJF with customer value and domain-specific components* could realise customer value as an objective measure to base decisions on and fix issues of WSJF's default configuration. Domain-specific needs which were found to be problematic in the current prioritisation were regulatory and compliance items, which precede others in development queue and are difficult to quantify due to their mandatory nature. In their simplest form, these two can be mere influencers of the Cost of Delay without exact definition of their weight, which however would not provide full objectivity. As interviewees considered business revenue to relate to customer value, a more precise component outside CoD reflecting revenue and customer experience through a measure such as NPS would not provide extra insight and would add to the complexity of prioritisation.

The company should also *consider the head start that its Finnish country unit has had in scaling agility*. The interviewees recognised that the initial stages of SAFe's implementation in the other country units were hindering the progress and maturity that had been achieved in Finland as the implementation of common practices from early-stage implementers has replaced adaptations in the Finland's model. The existing adaptations are the result of iterations of the development model since its initial implementation and therefore can be considered to be in place for a reason. To roll back this progress in the name of a common way of working across different countries would contradict the core message of the methods, which is to create adaptations based on surrounding factors (Moe et al., 2021). In other words, countries need their own adaptations to be successful, which increases the importance of the trust that managers must have in the people and the model for them to flourish.

Perhaps other country units should look more to Finland as more of an example of how to implement of SAFe, and common tools and platforms should incorporate adaptations which have been found beneficial in the leading SAFe country. Then again it is the responsibility and benefit of the implementing unit to figure out the best guidelines and customisations for itself. Common ways of working are also understandably sought after, but where the correct balance between individual customisation and the common good lies is a difficult, but necessary issue to establish.

On another issue of alignment, the current balance between team-level autonomy and planning activities in development was considered quite appropriate by the interviewees. However, one can still wonder whether the current balance will meet the demands of the future as the world becomes more uncertain. Given how the world has changed over the past years, and the vital role that telco industry plays in the modern society, *keeping this balance under review to meet future challenges* with corresponding agility would certainly do no harm.

10.2 Contributions for research

While this study has leaned towards practical contributions, many areas of research benefit from its findings. Firstly, as many existing studies focus on the early steps of adopting scaled agile methods and the challenges of this process, this thesis provides a viewpoint from an organisation that has achieved maturity in its SAFe-based scaled agile development model. The results of the empirical findings show that when this transformation has progressed from growing pains, the issues the adopting organisation may face become more domain-centric and reflect the nature and needs of its business environment.

The target company operates in a highly competitive market with strong regulation and customer orientation which provides a unique reference for the research of scaling agile. The company's operating environment shapes its development needs and requirements, which the need for high customer feedback demonstrates. Other domains are likely to have their own characteristics, which future scaled agile research could aim to uncover with more specialisation. The current generalisable research has its place in providing a solid base, but increased specialisation could yield new details from contexts with unique adaptations. In this thesis, the strong customer focus alongside security and regulatory requirements were identified as defining influencers of the development model's maturation.

The optimal alignment between team-level autonomy which enables the benefits of agility and planning in managing complexity are perhaps the defining issue of scaling agility (Conboy & Carroll, 2019; Moe et al., 2021). This study has shown that full agility through autonomy is not always the desired goal of an organisation. It is easy to think that methods always aim for full agility when in reality as uncovered in this study, this is not the case. Although interviewees perceived planning as the dominant side of this alignment, they felt it was quite suitable for the current development. This highlights the responsibility of adaptors to find out what works for them in their business environment. Exploring the diversity of this alignment is potential topic to consider in research as well.

Although the main academic contributions of this thesis lie in the area of scaled agile research, there are also contributions towards customer value research. Interviewees commented that the exact meaning of customer value is difficult to define and that environmental factors play a significant role in how people interpret it individually and in their environment. Alongside providing information for the issues of defining customer value, this study provides insight for customer value research in the context of agile development with a notable interest in its quantification as a basis for decision making. This angle has not attracted much research interest so far, which underlines its significance.

While research of agile software development and scaling agile is abundant (Uludağ et al., 2021), the same cannot be said of customer value's relation to scaling agile. Mentions of the need to include customer value aspects appear in research of agile prioritisation, such as Sambinelli & Borges (2019), but no

established research was found with a practical take on combining these two areas. In this sense, this work can be seen as an opening for future work on this topic to continue bridging the gap between (scaled) agile research and customer value. This area sees prominent practical benefits to ensure customer impact of development work and shares fundamental focus on customers. In addition to combining research of scaling agile and customer value, this study has also made initial efforts to quantify customer value, where work will hopefully continue in the future.

10.3 Limitations and directions for future research

Although this thesis has approached its subjects in a thorough and careful manner, there are topics and aspects which would benefit from a more focused look or a different angle. This section describes the limitations of this study in terms of the literature review and the empiric study, as well their methodology. These remarks increase the reliability and transparency of this thesis as an academic contribution with a strong practical interest.

While there are many studies of other methods than SAFe and comparison between them (Edison et al., 2022), they were left out from this thesis to prevent the scope from becoming too broad. The focus was concentrated on SAFe 5 because it is the most popular scaled method (*16th State of Agile Report*, 2022) and was the one the target company had based its model on. Still, covering other methods in detail could yield relevant information for improvement suggestions and possibly reveal fundamental differences on how other methods deal with different types of challenges.

This study serves as an example how a customer-oriented and established domain has shaped methodical requirements. Scaled agile research has not seen such industry specialisation, at least in a notable amount. Examining the characteristics of adaptations across different domains could provide relevant insights into their challenges and implementation. An example of a domain that could be expanded upon would be public organisations, such as government agencies. Despite having a dedicated version of SAFe for public organisations (Knaster & Leffingwell, 2020, p. 372) there has been very little research into its own characteristics and implementation. Studying this individual version of SAFe presents a potential scope for future academic endeavours.

The alignment between agility and forward planning was one of the core concepts of this study. This alignment issue offers many opportunities for future academic contributions expand upon, such as cultural characteristics. Exploring the underlining factors between different working cultures and their role and impact on scaled agile development would be an issue to extend scaled agile research to management studies. Would more hierarchic working cultures perhaps play along better with scaled methods that balance planning and autonomy in complex environments, whereas fully agile methods would only flourish in liberal contexts? This surfaced in the interviews but was not covered

in this work in detail. Another potential topic would be to study the impact that differences in this alignment impose on competitiveness and business performance, not to mention domain-specific characteristics and needs.

The description of customer value as a concept, its research history and its various dimensions also had to be rigorously scoped. In reading publications on the subject, it became clear that including more of the extensive but still fragmented research would have expanded the scope of this thesis beyond its main interests. A more nuanced approach with a broader focus on customer value could provide meaningful results given its fragmented state in research, which underlines comprehensive understanding of existing work. For example, value creation mechanisms could offer a relevant lens for studying (scaled) agility, which were not included in this study.

The diverse nature of organisations and their customers makes creating methods of quantifying customer value difficult, but a subject to hopefully receive research attention in the future. Given the differences in the meaning of value in customer value research alone (Zeithaml et al., 2020), creating methods is difficult to say the least. In this study, the proposed base is inspired by SAFe's WSJF method and its Cost of Delay component, which should reflect domain-specific needs. While this work is very initial, it provides basis for future work expand on. Resource integration as the source of value in B2B contexts described by Macdonald et al., (2016) would present another potential direction for research of customer value and agile methods to continue, especially in terms of outsourced resources and their effective management.

Even though the collection of the interviewees of this study was done representatively together with the supervisor from the target organisation, team-level developers were not interviewed. This is a clear drawback in the sampling, although the solid arguments for this decision were explained in the methodology section. The interviewees also argued that the interests of developers may not always be in line with those of the organisation, as they often work as external consultants rather than more committed in-house employees. In this sense, the decision to exclude developers from the sample of interviewees was justified, but still represents a shortcoming in terms of the representativeness of the interviewed personnel.

The previous sampling limitation also serves as a potential direction for future research. Examining the perceptions of developers could provide details not covered in this study. This would require careful management of the perspective of the study to ensure whose views are truly inspected and compared: the company's, the developers,' or their host organisation's, if developer are external by their nature. In addition to investigating this separate perspective, implementing, and testing the suggestions presented in this study for the target company would provide a practical subject to expand the work of this thesis in future. By examining the state before and after implementation of changes, a descriptive image of their efficacy could be tested. This could also reveal new directions for scaled agile to continue based on the results.

The author's own history in the target organisation also creates a limitation of its own. Although an effort has been made to maintain a neutral lens during the process of writing, humans are always prone to biases. One such example is the discussion of the SAFe implementation between the company's country units, where this thesis has examined the Finnish implementation. This makes the views susceptible to country unit bias as the interviewees were, albeit intentionally, gathered around a Finnish development segment. A similar study in the company from other country units could reveal practical information and provide insight for research of different levels of internal agility.

As with any study, it is subject to the limitations imposed by its human author. Although a comprehensive and careful approach was taken to the collection of relevant literature in line with the scoping decisions, there is always the possibility of having missed details or works which may or may not have proven significant for the end results. The same limitation also applies to the empirical material, the vast amount of which yielded relevant findings but also required intensive effort to analyse. Nevertheless, the contributions of this study, while not free from their limitations, stand on their own as a source for advancing knowledge of scaling agility at the crossroads with customer value research: the two of which have remained largely unconnected despite their common cornerstone, the customer.

REFERENCES

16th State of Agile Report. (2022). https://digital.ai/resource-center/analyst-reports/state-of-agile-report/

Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). *Agile software development methods: Review and analysis*. VTT Technical Research Centre of Finland.

Advanced Topic—What's new in the SAFe 5.1 Big Picture. (2021). *Scaled Agile Framework*. https://www.scaledagileframework.com/whats-new-in-the-safe-5-1-big-picture/

Afflerbach, P., & Frank, L. (2016). *Customer Experience Versus Process Efficiency: Towards an Analytical Framework About Ambidextrous BPM*. 22.

Alahyari, H., Berntsson Svensson, R., & Gorschek, T. (2017). A study of value in agile software development organizations. *Journal of Systems and Software*, *125*, 271–288. https://doi.org/10.1016/j.jss.2016.12.007

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. https://doi.org/10.18517/ijaseit.6.6.1374

Anwer, F., Aftab, S., Shah, S. S. M., & Waheed, U. (2017). *Comparative Analysis of Two Popular Agile Process Models: Extreme Programming and Scrum*. *8*(2), 7.

Ashmore, S., & Wedlake, M. (2016). Developing the Product Your Customer Really Wants: The Value of an Agile Partnership. *Information Resources Management Journal*, *29*(3), 1–11. https://doi.org/10.4018/IRMJ.2016070101

Beck, K. (1999a). *extreme programming eXplained: Embrace change*. Addison-Wesley.

Beck, K. (1999b). Embracing change with extreme programming. *Computer*, *32*(10), 70–77. https://doi.org/10.1109/2.796139

Beck, K., & Andres, C. (2004). *Extreme Programming Explained: Embrace Change, 2nd Edition* (2nd edition). Addison-Wesley.

Beck, K., Beedle, M., Bennekum, A. van, Cockburn, A., Cunningham, W., Fowler, M., James Grenning (last), Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for Agile Software Development*. http://agilemanifesto.org/

Becker, L., & Jaakkola, E. (2020). Customer experience: Fundamental premises and implications for research. *Journal of the Academy of Marketing Science*, *48*(4), 630–648. https://doi.org/10.1007/s11747-019-00718-x

Beedle, M., Devos, M., Schwaber, K., & Sutherland, J. (1998). *SCRUM: An extension pattern language for hyperproductive software development*.

Boehm, B., & Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *IEEE Software*, *22*(5), 30–39. https://doi.org/10.1109/MS.2005.129

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, *3*, 77–101. https://doi.org/10.1191/1478088706qp063oa

Cleland-Huang, J. (2015). Injecting Value-Thinking into Prioritization Decisions. *IEEE Software*, *32*(2), 14–18. https://doi.org/10.1109/MS.2015.44

Conboy, K., & Carroll, N. (2019). Implementing Large-Scale Agile Frameworks: Challenges and Recommendations. *IEEE Software*, *36*(2), 44–50. https://doi.org/10.1109/MS.2018.2884865

DeSantis, L., & Ugarriza, D. N. (2000). The concept of theme as used in qualitative nursing research. *Western Journal of Nursing Research*, *22*, 351–372. https://doi.org/10.1177/01939450022044467

Dingsøyr, T., Moe, N. B., Fægri, T. E., & Seim, E. A. (2018). Exploring software development at the very large-scale: A revelatory case study and research agenda for agile method adaptation. *Empirical Software Engineering*, *23*(1), 490–520. https://doi.org/10.1007/s10664-017-9524-2

Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, *85*(6), 1213–1221. https://doi.org/10.1016/j.jss.2012.02.033

Ebert, C., Abrahamsson, P., & Oza, N. (2012). Lean Software Development. *IEEE Software*, *29*(5), 22–25. https://doi.org/10.1109/MS.2012.116

Ebert, C., & Paasivaara, M. (2017). Scaling Agile. *IEEE Software*, *34*(6), 98–103. https://doi.org/10.1109/MS.2017.4121226

Edison, H., Wang, X., & Conboy, K. (2022). Comparing Methods for Large-Scale Agile Software Development: A Systematic Literature Review. *IEEE Transactions on Software Engineering*, *48*(8), 2709–2731. https://doi.org/10.1109/TSE.2021.3069039

Eklund, U., Holmström Olsson, H., & Strøm, N. J. (2014). Industrial Challenges of Scaling Agile in Mass-Produced Embedded Systems. In T. Dingsøyr, N. B. Moe, R. Tonelli, S. Counsell, C. Gencel, & K. Petersen (Eds.), *Agile Methods. Large-Scale Development, Refactoring, Testing, and Estimation* (pp. 30–42). Springer International Publishing. https://doi.org/10.1007/978-3-319-14358-3_4

Fenton, N., & Bieman, J. (2014). *Software Metrics: A Rigorous and Practical Approach, Third Edition*. CRC Press.

Figalist, I., Elsner, C., Bosch, J., & Olsson, H. H. (2019). Scaling Agile Beyond Organizational Boundaries: Coordination Challenges in Software Ecosystems. In P. Kruchten, S. Fraser, & F. Coallier (Eds.), *Agile Processes in Software Engineering and Extreme Programming* (pp. 189–206). Springer International Publishing. https://doi.org/10.1007/978-3-030-19034-7_12

Finkelstein, L. (1982). What is Not Measurable, Make Measurable. *Measurement and Control*, *15*(1), 25–32. https://doi.org/10.1177/002029408201500104

Fitzgerald, Kruschwitz, N., Bonnet, D., & Welch, M. (2014). Embracing Digital Technology: A New Strategic Imperative. *MIT Sloan Management Review*, *55*(2), 1-.

Fitzgerald, Stol, K.-J., O'Sullivan, R., & O'Brien, D. (2013). Scaling agile methods to regulated environments: An industry case study. *Proceedings of the 2013 International Conference on Software Engineering*, 863–872.

Gallarza, M. G., Gil-Saura, I., & Holbrook, M. B. (2011). The value of value: Further excursions on the meaning and role of customer value. *Journal of Consumer Behaviour*, *10*(4), 179–191. https://doi.org/10.1002/cb.328

Goldkuhl, G. (2012). Pragmatism *vs* interpretivism in qualitative information systems research. *European Journal of Information Systems*, *21*(2), 135–146. https://doi.org/10.1057/ejis.2011.54

Hannay, J. E., Benestad, H. C., & Strand, K. (2017). Benefit Points: The Best Part of the Story. *IEEE Software*, *34*(3), 73–85. https://doi.org/10.1109/MS.2017.66

Hilkka, M.-R., Tuure, T., & Rossi, M. (2005). Is Extreme Programming Just Old Wine in New Bottles: A Comparison of Two Cases. *Journal of Database Management*, *16*(4), 41–61. https://doi.org/10.4018/jdm.2005100103

Hirsjärvi, S., & Hurme, H. (2015). *Tutkimushaastattelu: Teemahaastattelun teoria ja käytäntö*. Yliopistopaino.

Hirsjärvi, S., Remes, P., & Sajavaara, P. (2009). *Tutki ja kirjoita* (13. osin uud. laitos). Tammi.

Hobbs, B., & Petit, Y. (2017). Agile Methods on Large Projects in Large Organizations. *Project Management Journal*, *48*(3), 3–19. https://doi.org/10.1177/875697281704800301

Hohl, P., Klünder, J., van Bennekum, A., Lockard, R., Gifford, J., Münch, J., Stupperich, M., & Schneider, K. (2018). Back to the future: Origins and directions of the "Agile Manifesto" – views of the originators. *Journal of Software Engineering Research and Development*, *6*(1), 15. https://doi.org/10.1186/s40411-018-0059-z

Holweg, M. (2007). The genealogy of lean production. *Journal of Operations Management*, *25*(2), 420–437. https://doi.org/10.1016/j.jom.2006.04.001

Horlach, B., Böhmann, T., Schirmer, I., & Drews, P. (2018). *IT Governance in Scaling Agile Frameworks*. 13.

Hsieh, H.-F., & Shannon, S. E. (2005). Three Approaches to Qualitative Content Analysis. *Qualitative Health Research*, *15*(9), 1277–1288. https://doi.org/10.1177/1049732305276687

Kalenda, M., Hyna, P., & Rossi, B. (2018). Scaling agile in large organizations: Practices, challenges, and success factors. *Journal of Software: Evolution and Process*, *30*(10), e1954. https://doi.org/10.1002/smr.1954

Kasauli, R., Knauss, E., Nilsson, A., & Klug, S. (2017, February 27). *Adding Value Every Sprint: A Case Study on Large-Scale Continuous Requirements Engineering*.

Klakegg, O. J., Williams, T., & Shiferaw, A. T. (2016). Taming the 'trolls': Major public projects in the making. *International Journal of Project Management*, *34*(2), 282–296. https://doi.org/10.1016/j.ijproman.2015.03.008

Klünder, J., Schmitt, A., Hohl, P., & Schneider, K. (2017). *Fake News: Simply Agile*. Gesellschaft für Informatik, Bonn. http://dl.gi.de/handle/20.500.12116/4891

Knaster, R., & Leffingwell, D. (2020). *SAFe 5.0 Distilled: Achieving Business Agility with the Scaled Agile Framework*. Addison-Wesley Professional.

Kreuzer, T., Röglinger, M., & Rupprecht, L. (2020). Customer-centric prioritization of process improvement projects. *Decision Support Systems*, *133*, 113286. https://doi.org/10.1016/j.dss.2020.113286

Kupiainen, E., Mäntylä, M. V., & Itkonen, J. (2015). Using metrics in Agile and Lean Software Development – A systematic literature review of industrial studies. *Information and Software Technology*, *62*, 143–163. https://doi.org/10.1016/j.infsof.2015.02.005

Laanti, M. (2014). Characteristics and Principles of Scaled Agile. In T. Dingsøyr, N. B. Moe, R. Tonelli, S. Counsell, C. Gencel, & K. Petersen (Eds.), *Agile Methods. Large-Scale Development, Refactoring, Testing, and Estimation* (pp. 9–20). Springer International Publishing. https://doi.org/10.1007/978-3-319-14358-3_2

Larman, C., & Basili, V. R. (2003). Iterative and incremental developments. A brief history. *Computer*, *36*(6), 47–56. https://doi.org/10.1109/MC.2003.1204375

Lehnert, M., Linhart, A., & Röglinger, M. (2016). Value-based process project portfolio management: Integrated planning of BPM capability development and process improvement. *Business Research*, *9*(2), 377–419. https://doi.org/10.1007/s40685-016-0036-5

Levy, Y., & J. Ellis, T. (2006). A Systems Approach to Conduct an Effective Literature Review in Support of Information Systems Research. *Informing Science: The International Journal of an Emerging Transdiscipline*, *9*, 181–212. https://doi.org/10.28945/479

Lindvall, M., Muthig, D., Dagnino, A., Wallin, C., Stupperich, M., Kiefer, D., May, J., & Kähkönen, T. (2004). Agile software development in large organizations. *Computer*, *37*(12), 26–34. Scopus. https://doi.org/10.1109/MC.2004.231

Livermore, J. A. (2008). Factors that significantly impact the implementation of an agile software development methodology. *Journal of Software*, *3*(4), 31–36. Scopus. https://doi.org/10.4304/jsw.3.4.31-36

Macdonald, E. K., Kleinaltenkamp, M., & Wilson, H. N. (2016). How Business Customers Judge Solutions: Solution Quality and Value in Use. *Journal of Marketing*, *80*(3), 96–120. https://doi.org/10.1509/jm.15.0109

Miller, G. G. (2001). The Characteristics of Agile Software Processes. *Proceedings of the 39th Int'l Conf. and Exhibition on Technology of Object-Oriented Languages and Systems (TOOLS'01)*, 3.

Misra, S. C., Kumar, V., & Kumar, U. (2010). Identifying some critical changes required in adopting agile practices in traditional software development projects. *International Journal of Quality and Reliability Management*, *27*(4), 451–474. Scopus. https://doi.org/10.1108/02656711011035147

Moe, N. B., Šmite, D., Paasivaara, M., & Lassenius, C. (2021). Finding the sweet spot for organizational control and team autonomy in large-scale agile software development. *Empirical Software Engineering*, *26*(5), 101. https://doi.org/10.1007/s10664-021-09967-3

Moreira, M. E. (2017). *The Agile Enterprise*. Apress. https://doi.org/10.1007/978-1-4842-2391-8

Myers, M. D. (1997). Qualitative Research in Information Systems. *MIS Quarterly*, *21*(2), 241. https://doi.org/10.2307/249422

Myers, M. D., & Newman, M. (2007). The qualitative interview in IS research: Examining the craft. *Information and Organization*, *17*(1), 2–26. https://doi.org/10.1016/j.infoandorg.2006.11.001

OED Online. (2022). Qualitative, adj. In *OED Online*. Oxford University Press. https://www.oed.com/view/Entry/155875

Olsson, H. H., & Bosch, J. (2014). From Opinions to Data-Driven Software R&D: A Multi-case Study on How to Close the 'Open Loop' Problem. *2014 40th EUROMICRO Conference on Software Engineering and Advanced Applications*, 9–16. https://doi.org/10.1109/SEAA.2014.75

Olsson, H. H., & Bosch, J. (2015). Towards continuous validation of customer value. *Scientific Workshop Proceedings of the XP2015*, 1–4. https://doi.org/10.1145/2764979.2764982

Paasivaara, M. (2017). Adopting SAFe to Scale Agile in a Globally Distributed Organization. *2017 IEEE 12th International Conference on Global Software Engineering (ICGSE)*, 36–40. https://doi.org/10.1109/ICGSE.2017.15

Paasivaara, M., Behm, B., Lassenius, C., & Hallikainen, M. (2018). Large-scale agile transformation at Ericsson: A case study. *Empirical Software Engineering*, *23*(5), 2550–2596. https://doi.org/10.1007/s10664-017-9555-8

Penkert, A., Steffens, J., & Krpanic, S. (2019). Making the Difference for Customers: Three Levers to Achieve Superior Experience in Telco Business! In P. Krüssel (Ed.), *Future Telco: Successful Positioning of Network Operators in the Digital Age* (pp. 275–283). Springer International Publishing. https://doi.org/10.1007/978-3-319-77724-5_24

Peppers, D., & Rogers, M. (2017). *Managing customer experience and relationships: A strategic framework* (Third edition). Wiley.

Petersen, K., & Wohlin, C. (2009). A comparison of issues and advantages in agile and incremental development between state of the art and an industrial case. *Journal of Systems and Software*, *82*(9), 1479–1490. https://doi.org/10.1016/j.jss.2009.03.036

Poppendieck, M., & Poppendieck, T. (2003). *Lean software development: An agile toolkit* (Nachdr.). Addison-Wesley.

Power, B. (2011, April 21). Customer-Centric Continuous Improvement. *Harvard Business Review*. https://hbr.org/2011/04/continuously-improving-custome

Ranganath, P. (2011). Elevating Teams from 'Doing' Agile to 'Being' and 'Living' Agile. *Proceedings - 2011 Agile Conference, Agile 2011*. https://doi.org/10.1109/AGILE.2011.40

Reichheld, F. F. (2003). The One Number You Need to Grow. *Harvard Business Review*.

Reichheld, F. F., & Markey, R. (2011). *The Ultimate Question 2.0: How Net Promoter Companies Thrive in a Customer-driven World*. Harvard Business Press.

Reifer, D. J., Maurer, F., & Erdogmus, H. (2003). Scaling agile methods. *IEEE Software*, *20*(4), 12–14. https://doi.org/10.1109/MS.2003.1207448

Ries, E. (2011). *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses* (1st Edition). Currency.

Rigby, D. K., Sutherland, J., & Noble, A. (2018). Agile at scale: How to go from a few teams to hundreds. *Harvard Business Review : HBR*, *96*(3).

Robson, C., & McCartan, K. (2016). *Real world research: A resource for users of social research methods in applied settings* (Fourth Edition). Wiley.

Rodríguez, P., Mäntylä, M., Oivo, M., Lwakatare, L. E., Seppänen, P., & Kuvaja, P. (2019). Chapter Four—Advances in Using Agile and Lean Processes for Software Development. In A. M. Memon (Ed.), *Advances in Computers* (Vol. 113, pp. 135–224). Elsevier. https://doi.org/10.1016/bs.adcom.2018.03.014

Salo, O., & Abrahamsson, P. (2008). Agile methods in European embedded software development organisations: A survey on the actual use and usefulness of Extreme Programming and Scrum. *IET Software*, *2*(1), 58. https://doi.org/10.1049/iet-sen:20070038

Sambinelli, F., & Borges, M. A. F. (2019). Strategies to Increase Customer Value in Agile Software Development. In G. S. Tonin, B. Estácio, A. Goldman, & E. Guerra (Eds.), *Agile Methods* (Vol. 981, pp. 63–79). Springer International Publishing. https://doi.org/10.1007/978-3-030-14310-7_5

Scaled Agile Inc. (2021). Posters. *Scaled Agile Framework*. https://www.scaledagileframework.com/posters/

Schwaber, K. (2004). *Agile project management with Scrum*. Microsoft Press.

Schwaber, K., & Beedle, M. (2002). *Agile software development with Scrum*. Prentice Hall.

Šmite, D., Moe, N. B., Šāblis, A., & Wohlin, C. (2017). Software teams and their knowledge networks in large-scale software development. *Information and Software Technology*, *86*, 71–86. https://doi.org/10.1016/j.infsof.2017.01.003

Squire, B., Readman, J., Brown, S., & Bessant, J. (2004). Mass customization: The key to customer value? *Production Planning & Control*, *15*(4), 459–471. https://doi.org/10.1080/0953728042000238755

Storti, A., & Clear, T. (2020). *The Contradiction of Agile Measures: Customer as Focus, but Process as Measured?* 13.

Takeuchi, H., & Nonaka, I. (1986, January 1). The New New Product Development Game. *Harvard Business Review*. https://hbr.org/1986/01/the-new-new-product-development-game

Tamm, T., Seddon, P. B., Shanks, G., & Reynolds, P. (2011). How Does Enterprise Architecture Add Value to Organisations? *Communications of the Association for Information Systems*, *28*. https://doi.org/10.17705/1CAIS.02810

The SAFe Delusion – Information for decision-makers considering the SAFe framework. (2022). https://safedelusion.com/

Uludağ, Ö., Putta, A., Paasivaara, M., & Matthes, F. (2021). Evolution of the Agile Scaling Frameworks. In P. Gregory, C. Lassenius, X. Wang, & P. Kruchten (Eds.), *Agile Processes in Software Engineering and Extreme Programming* (pp. 123–139). Springer International Publishing. https://doi.org/10.1007/978-3-030-78098-2_8

Valentina, F. N., Antoneta, T. D., & Anişoara, D. (2018). Enabling Customer-Centricity and Relationship Management using Net Promoter Score. *Valahian Journal of Economic Studies*, *9*(2), 115–126.

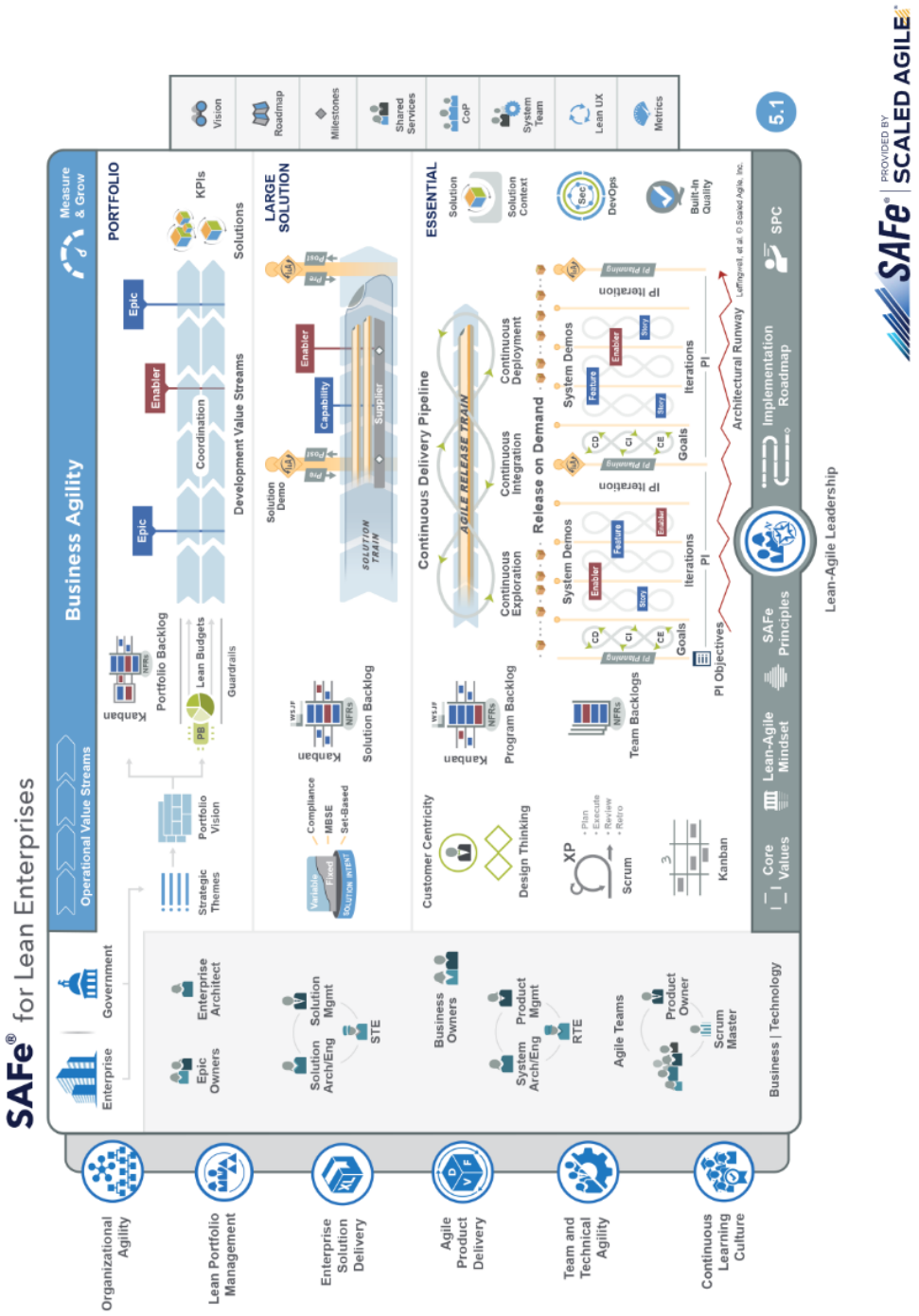
van Wessel, R. M., Kroon, P., & de Vries, H. J. (2021). Scaling Agile Company-Wide: The Organizational Challenge of Combining Agile-Scaling Frameworks and Enterprise Architecture in Service Companies. *IEEE Transactions on Engineering Management*, 1–14. https://doi.org/10.1109/TEM.2021.3128278

Woodruff, R. B. (1997). Customer value: The next source for competitive advantage. *Journal of the Academy of Marketing Science*, *25*(2), 139. https://doi.org/10.1007/BF02894350

Zeithaml, V. A., Verleye, K., Hatak, I., Koller, M., & Zauner, A. (2020). Three Decades of Customer Value Research: Paradigmatic Roots and Future Research Avenues. *Journal of Service Research*, *23*(4), 409–432. https://doi.org/10.1177/1094670520948134

APPENDIX 1 – SAFE 5.1 BIG PICTURE

Fetched from posters by Scaled Agile Inc. (2021)



APPENDIX 2 – ENGLISH INTERVIEW BODY

Theme 0: Background information

1. How many years of experience do you have of working around SAFe or other agile method(s), such as LeSS, Spotify (tribes) or Scrum?

2. Where do you work in terms of SAFe's structure in the organisation and what is your role?
    a. *E.g., level team, train, portfolio*

Theme 1: Challenges of current SAFe configuration (RQ1.1)

3. What is your overall perception of SAFe and its functioning in your position/role?

4. What challenges do you see or have experienced in the current SAFe configuration in your position?
    a. What factors create or affect these challenges?
    b. How do they influence development and customer outcome?

5. What upsides do you see or have experienced in the current SAFe configuration in your position?

6. What grade would you give to the current configuration of SAFe from 0 to 5 regarding its overall functioning? 0 is the lowest and 5 is the best.

7. What actions would you propose to solve or ease challenges and promote upsides?

A core issue in scaling agile is finding the optimal balance between team-level autonomy and managerial forward planning.
Agile development's benefits are commonly thought to root from the autonomy of allowing agile teams to make the best decisions with results of iterative and incremental development. On the other hand, large enterprises manage their inherent complexity with forward planning activities, which can be viewed to contradict with the spirit of agility.

8. How would you rate the current status of this balance in your position?
    a. 0 means that forward planning defines teams' duties and tasks completely without them being able to make or effect decisions. 5 means that teams possess autonomy to make all decisions.

Theme 2: Customer value in current SAFe configuration (RQ2.1 and 2.2)

Robert Woodruff has defined customer value followingly:

> *"Customer value is a customer's perceived preference for and evaluation of those product attributes, attribute performances, and consequences arising from use that facilitate (or block) achieving the customer's goals and purposes in use situations".*

9. Does this definition match the one you have in your position?
    a. Does it reflect the general meaning of customer value in discussions in the organisation?
    b. How would you define customer value based on working in the organisation?

10. How do you see customer value to be present in the current SAFe development process where you work?
    a. How are development items currently prioritised on the level/role of SAFe you work at?
        i. Is prioritisation based on customer value?
    b. How is customer value measured and tracked in the current development model?
        i. How do you see that it could be measured more accurately?
    c. How would you grade the current SAFe configuration in terms of how it incorporates and prioritises development based on customer value? 0 means customer value does have any role and 5 means it has all the say without any other influencing factors.

11. What actions would increase customer value's role in development model?
    a. How about in development item prioritisation?

12. How do you see the balancing between complexity from customer specific customisation and customer value (feedback/wishes) to affect development in the organisation?
    a. How would you rate the status of this balance on a scale from 0 to 5? 0 means that targeting simplicity is the deciding factor and 5 that customer value.

APPENDIX 3 – FINNISH INTERVIEW BODY

Teema 0: Taustatiedot

1. Kuinka monta vuotta sinulla on kokemusta työskentelystä SAFen tai muiden ketterin kehitysmenetelmien, kuten LeSS:n, Spotifyn (heimot) tai Scrumin parissa?

2. Missä työskentelet SAFen rakennetta ajatellen ja mikä on roolisi?
    b. *Esim. team, program, portfolio*

Teema 1: Nykyisen SAFe-konfiguraation haasteet (TK1.1)

3. Mikä on yleiskuvasi SAFesta ja sen toiminnasta roolissasi?
    a. Minkä arvosanan antaisit tämänhetkiselle SAFe-konfiguraatiolle sen yleisen toimivuuden perusteella? 0 tarkoittaa heikointa ja 5 parasta.

4. Millaisia haasteita näet tai olet kokenut nykyisessä SAFe-konfiguraatiossa roolissasi?
    a. Mitkä tekijät luovat tai vaikuttavat näihin haasteisiin?
    b. Miten ne vaikuttavat kehitykseen ja asiakaslopputulemaan/vaikutukseen?

5. Millaisia hyviä puolia tai haasteita näet nykyisessä SAFe-konfiguraatiossa roolissasi?

6. Millaisia toimenpiteitä ehdottaisit haasteiden ratkaisemiseksi ja hyvien puolien edistämiseksi?

Keskeinen haaste ketteryyden skaalaamisessa on optimaalisen tasapainon löytäminen tiimitason autonomian ja johtovetoisen suunnitelmallisuuden välillä. Ketterän kehityksen etujen on yleisesti ajateltu juontuvan tiimitason autonomiasta, jonka myötä niiden jäsenet saavat tehdä parhaat päätökset yhdessä iteratiivisen ja inkrementaalisen kehityksen voimin. Toisaalta suuret yritykset hallitsevat niiden monimutkaisuutta suunnittelemalla toimintaansa etukäteen, minkä voi nähdä sotivan ketteryyden periaatteita vastaan.

7. Miten arvioisit tämän tasapainon nykytilaa roolisi ympärillä?
    a. Minkä arvosanan antaisit sille nollasta viiteen? 0 tarkoittaa, että suunnitelmallisuus määrittelee kaiken toiminnan etukäteen, eivätkä tiimiläiset voi vaikuttaa niihin. 5 tarkoittaa, että tiimiläisillä on täysi autonomia tehdä kaikki päätökset.

Teema 2: Asiakasarvo nykyisessä SAFe-konfiguraatiossa (TK2.1 ja 2.2)

Robert Woodruff määrittelee asiakasarvon seuraavasti:

> *”Asiakasarvo on asiakkaan kokema mieltymys ja arvio niistä tuotteen attribuuteista, niiden suoriutumisesta ja käytön seurauksista, jotka käyttötilanteissa edesauttavat tai estävät asiakkaan tavoitteiden saavuttamista käyttötilanteissa”.*

> *"Customer value is a customer's perceived preference for and evaluation of those product attributes, attribute performances, and consequences arising from use that facilitate (or block) achieving the customer's goals and purposes in use situations."*

8. Vastaako tämä määritelmä käsitystäsi asiakasarvosta roolissasi?
    a. Vastaako se käsitystä asiakasarvosta yleisesti organisaatiossa?
    b. Miten määrittelisit asiakasarvon organisaatiossa yleisesti?

9. Miten asiakasarvoa mitataan ja seurataan nykyisessä kehitysmallissa?
    a. Miten sitä voisi mielestäsi mitata tarkemmin tai osuvammin?

10. Miten näet asiakasarvon olevan läsnä nykyisessä SAFe-kehitysmallissa missä työskentelet?
    a. Miten kehityskohteita tällä hetkellä priorisoidaan toimenkuvassasi?
        1. Perustuuko priorisointi asiakasarvoon?
    b. Minkä arvosanan antaisit nykyiselle SAFe-konfiguraatiolle sen suhteen, kuinka se huomioi ja priorisoi kehitystä asiakasarvon perusteella? 0 tarkoittaa, että asiakasarvolla ei ole mitään sanavaltaa ja 5 että se määrittää kaiken ilman muita vaikuttavia tekijöitä.

11. Millaiset toimet kasvattaisivat asiakasarvon roolia kehitysmallissa?

12. Miten näet tasapainoilun asiakaskohtaisen kustomoinnin tuottaman monimutkaisuuden ja asiakasarvon (esim. asiakastoiveet tai palaute) välillä vaikuttavan kehitykseen organisaatiossa?
    a. Miten arvioisit tämän kaksinaisuuden nykytilaa asteikolla 0-5? 0 tarkoittaa, että yksinkertaisuuden tavoittelu on määräävässä asemassa ja 5 että asiakasarvo.