NTNU
Norwegian University of
Science and Technology

Challenges for Autonomous Development Teams

A Multiple Case Study

Bjørn Dahl
Lina Sund Karlsen
Stine Schjødt-Osmo

Industrial Economics and Technology Management
Submission date: June 2018
Supervisor: Hanne Olofsson Finnestrand, IØT
Co-supervisor: Nils Brede Moe, SINTEF

Norwegian University of Science and Technology
Department of Industrial Economics and Technology Management

Problem Description

The use of autonomous teams has been a growing trend in several industries the recent years. This thesis aims to provide new insight into how challenges for autonomous development teams relate to autonomy. The research question is therefore:

What are the challenges for autonomous development teams, and how do they relate to autonomy?

We conduct a qualitative multiple case study with an inductive approach, in order to identify challenges in autonomous development teams, and how they are approached. Then, we couple the findings with theory to evaluate the effects on autonomy.

Preface

This master's thesis is the final work of our Master of Science in Industrial Economics and Technology Management within Strategic Change Management at NTNU. The thesis is written in collaboration with the A-team research project at SINTEF over the course of spring 2018.

The study explores challenges and autonomy within development teams in three case companies. We would like to thank the case companies for their openness and collaboration, and all the interviewees for their time spent contributing to our research.

We also owe a debt of gratitude to our academic supervisor Hanne O. Finnestrand. Your feedback, motivation and guidance through the research period have been invaluable. Special thanks go to our co-supervisor Nils B. Moe at SINTEF for sharing your oasis of knowledge, opening the doors to the A-team project and its participating companies, as well as for your availability and encouraging words along the way. We would also like to thank the other researchers in the A-team project. All of your contributions are highly appreciated.

Trondheim, 8th June 2018

Bjørn Dahl Lina S. Karlsen Stine Schjødt-Osmo

Bjørn Dahl Lina Sund Karlsen Stine Schjødt-Osmo

Abstract

Purpose: The purpose of this thesis is to identify and explore how challenges for autonomous development teams relate to autonomy. As the use of autonomous teams has seen a renewed interest in recent years, more knowledge of the implications for how to succeed with autonomous teams is considered valuable.

Method: Empirical data from development teams in finance and consulting companies is gathered through semi-structured interviews for an inductive multiple case study. We identify challenges from the empirical data through the following themes: Overall Direction, External Coordination, Intra-team Coordination, Decision-making and Human Factors. Theory from different theoretical aspects, including STS and agile development, is used to provide a thorough background and a basis for our discussions.

Findings: Eight challenges for autonomous development teams are identified: commitment to goals, external dependencies, coordination mechanisms, process improvement, unequally distributed decision-making authority, customer authority, empty role titles and specialists. Autonomy is viewed in terms of three aspects: the individual freedom of the members, shared decision-making in the team and authority given to the team from the external environment. All of the challenges relate to one or more aspects of autonomy. Our study contributes to the knowledge on autonomous development teams, which is valuable for both researchers in the field and industries engaged in the practice.

Sammendrag

Formål: Formålet med denne studien er å identifisere og utforske relasjonene mellom utfordringer og autonomi i autonome utviklingsteam. Ettersom interessen for autonome team har økt betraktelig de siste årene, anses mer kunnskap om hvordan lykkes med autonome team som et viktig bidrag.

Metode: Empirisk data fra utviklingsteam innen finans- og konsulentbedrifter er innhentet gjennom semi-strukturerte intervjuer til en induktiv multiple case-studie. Vi identifiserte utfordringer ut fra det empiriske materialet gjennom følgende temaer: Overordnet retning, ekstern koordinering, team-intern koordinering, beslutningstaking og menneskelige faktorer. Teori fra ulike teoretiske retninger, blant annet STS og agil utvikling, er benyttet for å gi en grundig teoretisk bakgrunn og et grunnlag for analyse og diskusjon.

Funn: Det er identifisert åtte utfordringer for autonome utviklingsteam: forpliktelse til mål, eksterne avhengigheter, koordineringsmekanismer, prosessforbedring, ujevn fordeling av beslutningsmyndighet, kundens myndighet, tomme rolletitler og spesialister. Autonomi blir utforsket gjennom tre aspekter: den individuelle friheten til medlemmer, delt beslutningstaking i teamet og myndighet gitt teamet fra eksterne aktører. Alle utfordringene viser seg å være koblet til ett eller flere av autonomiaspektene. Vår studie bidrar til kunnskap om autonome utviklingsteam, og anses verdifull for både forskning på feltet og industri der autonome team er av interesse.

Contents

List of Figures

List of Tables

1 Introduction

There has been a renewed interest in self-managing teams across various industries in recent years (Van der Vegt, Bunderson & Kuipers, 2014). Goodman, Devadas & Hughson (1988) define self-managing teams, or autonomous teams, as groups of individuals with interdependent tasks which they can self-regulate. Members of these teams are often cross-skilled and able to plan and execute every phase needed from initiation to completion of the desired product. The team members are responsible for themselves and their work, as opposed to the traditional manager-led teams (Morgan, 2006; Parker & Wall, 1998).

It has been shown that self-managing teams have many benefits. Manz & Sims (1987) underline the flexibility of self-managing teams, allowing them to respond rapidly to a changing environment. Furthermore, decision-making authority is decentralized and placed closer to the customer, speeding up processes because there is no need for approval from higher level managers (Gerwin & Moffat, 1997; Wageman, 1997). Another benefit is increased performance and job satisfaction among team members. Since they are responsible for their own work and how it should be carried out, they are more motivated and engaged (Parker & Wall, 1998; Stewart & Manz, 1995; Wageman, 1997).

Self-managing teams are a core tenet of the socio-technical approach (Goodman et al., 1988; Stewart & Manz, 1995, Parker & Wall, 1998; Van Amelsvoort, 2000). The socio-technical thinking has its roots in the 1950s, when organizational researchers began questioning the sole focus on either the technical aspect (Scientific Management) or the social aspect (Human Relations) of organizations (Van Eijnatten, 1993). Socio-technical thinking recognized the interaction between people and technology within organizations, and sought to combine the two aspects. The key point was to align the individual needs of the workers with the technological dimension to maximize organizational performance (Trist, 1981).

One of the industries in which the socio-technical systems (STS) and its practice of self-managing teams have resurged recently, is the software development industry (Nerur, Cannon, Balijepally & Bond, 2010). Today, so-called agile teams with agile practices are frequently used when developing software (Dingsøyr, Nerur, Balijepally & Moe, 2012). Nerur et al. (2010) argue that agile practices align well with the principles of STS. Agile methodologies emphasize an open and collaborative development environment where the developers possess the decision-making authority when assigning roles and deciding problem-solving strategies. According to Nerur et al. (2010), this resembles self-management as described in STS.

While the benefits of self-managing teams are well known, there are also challenges. Wageman & Fisher (2014) highlight making shared decisions as difficult, as the teams have to determine which members should participate in the decision-making. They also point to information flow as a challenge, since all of the team members share the responsibility of gathering and spreading relevant information within the team. Manz & Sims (1987) in Balkema & Molleman (1999) emphasize that team members need to learn new technical and social abilities if they are to succeed with self-management. Without technical competence they cannot do their jobs, and without social abilities they cannot communicate properly. For self-management in agile teams specifically, Moe, Dingsøyr & Dybå (2009) identified team learning as a challenge. According to them, failure to learn is often due to teams not putting aside enough time for process improvement. Moe et al. (2009) also point to individual commitment as a challenge. This entails team members pursuing their own goals instead of the team’s goals. In other words, there are challenges both on the team and individual level (Moe et al., 2009).

In this thesis, we will refer to the term *challenge* as a matter that impedes the team's processes. The teams we study are agile development teams. These teams fall under the category of problem-solving teams, described by Ratliff, Beckstead & Hanks (1999) as teams that combine a variety of skills and knowledge to solve problems with unclear structure and boundaries. Task-solving often entails exploring new areas and producing something unique; not doing standard work to solve routine tasks. We will therefore refer to them as *autonomous development teams*. Hence, our research question (RQ) is:

What are the challenges for autonomous development teams, and how do they relate to autonomy?

We will identify the challenges using an inductive approach. They are based on the empirical findings of our multiple case study. Subsequently, we will look at how the challenges relate to autonomy. We use the terms autonomy and autonomous teams interchangeably with self-management and self-managing teams. All of the terms used in the RQ, as well as the category of teams we are studying, will be elaborated upon in Section 2.3.

Before answering the RQ, we will provide a theoretical background. Moving on, we expand on the research method and present our case companies, followed by the empirical analysis. We then discuss our findings. Finally, we conclude our research and provide an agenda for future research. The structure of our thesis can be seen in Figure 1.1.

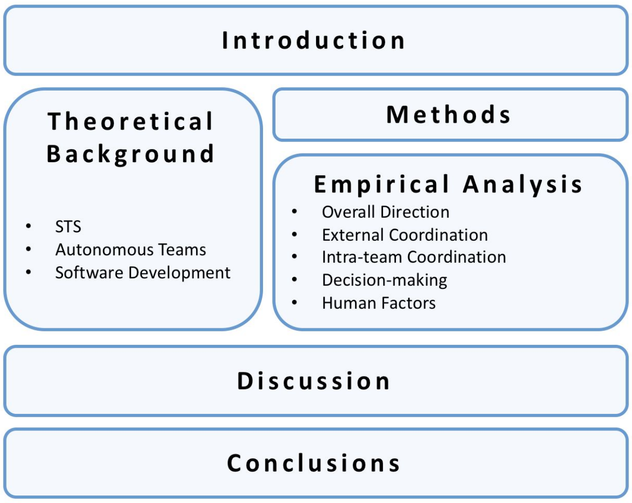


Figure 1.1: Structure of the thesis.

2 Theoretical Background

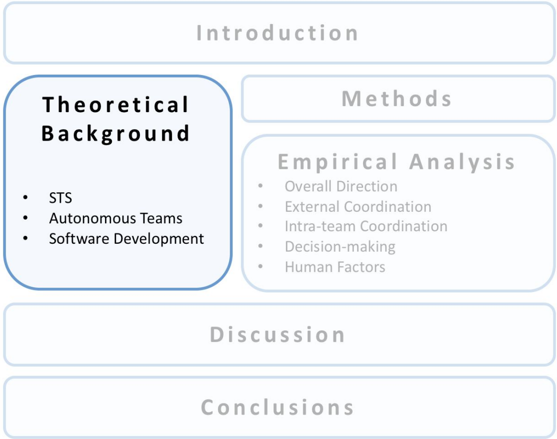


Figure 2.1: Structure of the theoretical background.

In order to answer our research question concerning how challenges for autonomous development teams relate to autonomy, a theoretical background on autonomous teams is needed. Theory from different theoretical aspects is included in order to provide a thorough theoretical foundation. We begin by giving a historical background of how autonomous teams arose, and how they became a widespread practice. Subsequently, we challenge autonomy by exploring literature criticizing autonomous teams. Finally, we look into how autonomous teams found their way into software development, before this chapter ends with necessary definitions and explanations for the research question. The content of this chapter is illustrated in Figure 2.1.

2.1 Socio-technical Systems and Autonomous Teams

In the early 1900s, classical management theory argued for detailed planning and control in order to maximize productivity and efficiency. Scientific Management and its lead proponents attempted to establish universal principles for management and organizations (Watson, 2008). One of these principles was dividing an organization into mutually exclusive departments where workers were specialized. The departments were connected through a single chain of command. Every employee had someone to answer to, this someone had another someone, and so the chain of command continues (Morgan, 2006).

Throughout the century, Scientific Management's ideas were challenged by other approaches. Critics of Scientific Management claimed that the social dimension, the employees, also had their place in management theory. Under the leadership of Harvard University professor Elton Mayo, the Hawthorne studies in the 1920s and 1930s recognized the importance of social needs in the workplace (Morgan, 2006). What later became known as Human Relations paid great interest in giving the employees meaningful work, recognition and responsibility, in an attempt to increase their involvement and usefulness in the organization (Herzberg, 1966).

Moving on to the 1950s, some organizational researchers suggested that relying solely on either the technical aspect or the social aspect of the organization was not enough. The new approach, known as Socio-technical Systems (STS), sought to merge the technical with the social aspect of the organization (Trist, 1981). To jointly optimize the technical and social aspects, Cherns (1976) presented nine principles for socio-technical design. The principles systemized earlier writings by Emery & Trist (1972) and Herbst (1974). One of these principles is minimal critical specification. This entails that you should not plan more than you have to; if you do plan, you close options that otherwise could have been left open.

Supporters of STS also recognized the interdependencies between people, technology and environment (Cummings, 1978). An example is Emery & Thorsrud (1976), who argued that workers should be able to participate in organizational decisions. Without involvement, they claimed that the workers would never consider themselves part of a democracy. Trist (1981) shared this view, and encouraged a more horizontal and participative organization, moving away from the vertical orientation of Scientific Management.

Autonomous Teams in STS

Van Eijnatten (1993) follows up Trist's (1981) point by proposing that organizations should structure their work design such that it maximizes human resources. To illustrate, workers both designing and solving problems result in better performance and more satisfying work. Stewart & Manz (1995) underline the importance of doing this together through a shift from individual to group work methods. This means putting the workers together in teams with more control over their own tasks and work routines.

According to Goodman et al. (1988) and Parker & Wall (1998), there is a wide range of terms used for this type of work design[1]. In this thesis we will use the terms *autonomous teams* and *self-managing teams* interchangeably. Parker & Wall (1998) argue that the defining feature of autonomous teams is employees having discretion over the continuously operational decisions related to who does what, when and how. A key point is that their tasks are interdependent, and that they together form a whole product or service (Van Amelsvoort, 2000). Also, the team is generally expected to cover every function it needs to fulfill its goals, which favors a high and cross-functional competence level (Morgan, 2006; Parker & Wall, 1998).

In line with this, Goodman et al. (1988) define autonomous teams as groups of individuals with interdependent tasks they can self-regulate. As opposed to a traditional work group led by a manager, a self-managing team is responsible for controlling its own processes and set of tasks. The essential elements include face-to-face interaction, a physically defined work area, a set of interdependent tasks and members having control (authority and responsibility) over the management (planning, organizing and monitoring) and execution of the tasks (Goodman et al., 1988). We will be using this definition in our thesis, as it covers the most important characteristics of autonomous teams.

[1] Examples are semi-autonomous work groups, self-managing teams, flexible work groups, high-performance work teams, self-directed work groups and self-designing or self-leading teams.

There are many studies exploring the benefits of autonomous teams. Stewart & Manz (1995) claim that empowerment can be linked to the intrinsic motivation of the individuals, as deciding on own job tasks makes one more willing to engage to the work. In turn, this may lead to higher job satisfaction and better organizational performance. Gerwin & Moffat (1997) follow up this point, and argue that autonomous teams can help better meet technical objectives and speed up team processes. With more autonomy, the teams can optimize the design of the products themselves, without having to wait for approvals from higher organizational levels. Additionally, there are case studies showing productivity, technical measurements, product quality, job satisfaction and general attitudes among workers to improve significantly over time working in self-managing teams (Cummings & Worley, 2015; Parker & Wall, 1998).

Another benefit is the structure of an autonomous team in a shifting environment. Manz & Sims (1987) claim that self-managing teams is a useful approach in meeting the increasing interdependence, complexity and uncertainty organizations face from their environment. They refer to Susman (1976) and state that a team can reallocate resources for variances in the work conditions more effectively compared to individuals in a rigid structure. As team members, they flexibly contribute to the team's assignment rather than commit to one specific job.

Supervision of Autonomous Teams

On the other hand, Stewart & Manz (1995) also point out that self-managing teams bring about challenges, as many organizations are unsuccessful in implementing them. They claim that self-management entails a reallocation of responsibility and authority, where the individuals' possibility to make decisions increase. The reallocation leads to new roles, and thereby new leadership roles. Langfred (2007) agrees with this, and argues that leaders of self-managing teams need to acquire new knowledge and skills to adjust to their new way of working.

On the same topic, Stewart & Manz (1995) discuss how important supervision is, as the leader now is expected to lead the team to lead itself. Their conclusion is the same as Wageman and Fisher's (2014); that leaders of self-managing teams need to shift from a mode of command, to a mode of support and facilitation.

In terms of the external environment, supervision has also been addressed by Cummings (1978). He argues that even though the teams are designed to take on tasks themselves, there is still need for external supervision to initiate and provide a framework for the teams. This is because he considers the supervisory role as essential in helping the team manage its boundaries, and for helping team members develop and organize themselves into an effective team. Mathieu, Gilson & Ruddy (2006) found that whereas there had been a shift towards working in autonomous teams, the external managers had not adapted. They insinuate that the external leaders simply had not yet committed to the new work design. According to Druskat & Wheeler (2004), the commitment from external leaders is critical to empower the self-managing teams, as they should constantly guide and develop their teams so that they become more independent. In other words, external leaders are responsible for getting the team to manage itself from the outside (Manz & Sims, 1987).

Managing Information

Another challenge is to manage the flow of information in the organizational design of autonomous teams. Lawler (1992) explains how information boards and electronic devices can be helpful in creating a horizontal flow of information in a decentralized organization. However, he also points out two problems.

The first problem is the difference between information and knowledge. Although people are informed, it does not mean they know what to do with the information. Hence, information is not the same as knowledge (Nonaka, 1994). Even though you put up a lot of information on a board, it does not mean that team members can make use of it (Lawler, 1992). The second problem is the concept of information overload. Too much information might result in the phenomenon described by Schick, Gordon & Haka (1990) as more information than individuals are able to process. There comes a point where more information is no longer useful, and rather becomes confusing and interruptive.

Making Shared Decisions

Next, a significant challenge autonomous teams face is shared decision-making (Wageman & Fisher, 2014). This is similar to shared leadership, which Carson, Tesluk & Marrone (2007) define as sharing decision-making authority among team members instead of assigning it to a designated leader. Hoegl & Parboteeah (2006) point out that team members have to accept joint decisions made in the team, and that there sometimes are compromises between polarized positions within the team. According to Wageman & Fisher (2014), a problem that often leads to decision-making failure is that it is difficult to determine and gather the information relevant for the decision at hand. Determining who should participate in the decision-making process is also difficult (Wageman & Fisher, 2014). Hackman (1987) argues that teams often struggle to identify which members hold the necessary expertise to solve the task at hand. He states that the difficulty can be associated with team members giving credence to task-irrelevant considerations.

The difficulty of shared decision-making underline that the implementation of self-managing teams is not a one-step manner. Manz & Sims (1987) in

Balkema & Molleman (1999) emphasize that team members need to learn technical skills and social abilities in order to become successfully self-managed. Without technical skills they cannot do their job, and without social abilities they cannot communicate and learn collectively. They stress that teams will not achieve self-management without training and support. Cummings (1978) shares this view, and states that especially social aspects such as decision-making and team dynamics are created through members working together and adjusting to their environment.

According to Pearce III & Ravlin (1987), lack of commitment to a shared set of goals may lead to team members feeling less motivated, and to an unclear sense of direction, since they are not on the same path. There are several benefits of letting team members participate in setting the goals. Hackman & Oldham (1980) argue that a higher degree of participation leads to increased motivation and sense of empowerment. They point to three important factors for motivation: making people responsible for outcomes of their work, giving them feedback on the work performance, and ensuring that the work is meaningful. Hackman (1987) argues that the meaningfulness of goals as important, since doing trivial work, or work that does not fit the individual's set of values, will not increase motivation. Also, Latham & Saari (1979) state that participation in goal setting can lead to a better understanding of what it takes to achieve the goals, as the road towards the goal will be easier to identify and follow.

The Possibility of Autonomy

Balkema & Molleman (1999) point out another challenge, in that leaders can struggle with the notion of self-management itself. If the leaders' impression is that the team cannot control their own processes, they might feel the need to intervene. Resistance to self-management among leaders might be due to leaders believing that the team is unable to manage itself. Another view is

Wageman & Fisher’s (2014), who propose that the resistance might indicate that leaders are not willing to withdraw their control of the team.

The resistance is not limited to leaders though. According to Wageman & Fisher (2014), team members in general might show unwillingness in adapting to self-management. Wageman (1997) points out that in many teams that are supposed to be self-managed, team members continue to work individually without engaging in collective decision-making and problem-solving. Tata & Prasad (2004) claim that team members need to genuinely affect managerial decisions in order to benefit from self-management. Otherwise, they will only experience symbolic self-management. If the managerial decisions are only affected by symbolic input, the team members might hesitate in embracing self-management. This line of thinking is the reason why many researchers have raised the question of whether or not a team can become truly self-managed (Balkema & Molleman, 1999; Stewart & Manz, 1995; Wageman & Fisher, 2014).

Taggar, Hackew & Salah (1999) state that the process of role making and role taking may lead to the development of an informal hierarchy of relationships within the team. Hackman (1987) suggests that this process should be left to the team and its members, as a discussion of leadership roles in a team should, and will, happen naturally. As this discussion is important to the team’s development process, the leadership structure of a team should rarely be decided in advance.

2.2 Software Development

One of the industries in which self-managing teams are present today is the software development industry (Nerur et al., 2010).

What is Software Development?

Software development is the process of planning, designing, implementing and maintaining software (Boehm, 1988). To determine the order of the stages of the process, and what it takes to complete them, software developers use a software process model. Primarily, the function of such a model is to provide guidance on what to do next, and for how long. Since the beginning of software development, these models have evolved from simple steps to complex phases (Boehm, 1988).

The early, basic models essentially had two stages. You wrote some code, and then you tried to fix it. Over time, the code became unstructured, difficult to sort through and hard to test properly. As a result, software developers saw the need of a design stage and a testing stage. Also, it became eminent that some sort of feedback between the stages was needed. This resulted in the waterfall model, which quickly became the standard for developing software (Boehm, 1988). The waterfall model was heavily plan-based. Every work activity was planned in detail before being carried out. This line of thinking resembles principles of Scientific Management, and it is no surprise that the waterfall model was heavily based on the same paradigm (Kakar, 2014; Nerur et al., 2010).

Practices in Agile Software Development

The same way Scientific Management has been challenged by STS, the waterfall model has lost ground to less plan-based approaches (Kakar, 2014; Nerur et al., 2010). Today, software requirements and customer demands are subjects to change, often making extensive preplanning a waste of time. Less plan-based approaches allow for more flexibility and responsiveness when dealing with complex problems in a more ambiguous and fast-paced world (Dybå, 2000; Nerur & Balijepally, 2007). These approaches are today known as agile models, processes or methods. Instead of attempting to plan and optimize the entire process ahead of time, an agile process commits to only plan what has to be planned. By doing this, you combine the notion of traditional planning with the ability to adapt to changes (Gren, 2017).

The core practice of an agile software development process is to deliver value in short intervals. Customers are involved at an early stage, letting the feedback from one interval influence the next (Gren, 2017). Being this responsive to changes places great importance on the team's competence and its decentralized decision-making (Evans & Davis, 2005). According to Cockburn & Highsmith (2001), without self-management, the structure of the team will be too hard-lined to enable such rapid responses. Similar to STS, agile software development processes also rely on self-managing teams, letting the workers closest to the customers organize the work (Boehm & Turner, 2003; Nerur & Balijepally, 2007). Hence, when referring to agile teams in this thesis, we consider them as self-managed.

Agile teams usually consists of many, not necessarily rigid, roles (Gren, 2017). This is important, because agile teams are, similar to how Morgan (2006) and Parker & Wall (1998) describe autonomous teams earlier presented, usually responsible for covering every function it needs. Hence, there are many perspectives that need attention in ensuring an effective software development process. Table 2.1 shows an overview of roles you typically find in an agile software development team (Beck & Andres, 2004).

Table 2.1: Roles in agile development teams.

Role	Responsibility
Developer / Programmer	Writes the main parts of the code. In some cases divided into front-end (works on what the user can see) and back-end (works on what is running in the background) developers.
Tester	Works solely on testing code that is ready (users of the software might also be testers).
Architect	Oversees the software on a system-level and how different parts of the program communicate with each other.
Interaction Designer	Focuses on user experience and what the software is trying to accomplish for the user.
Project Manager / Product Owner	Initiates the project and coordinates the team internally and/or externally.

In order to make all of these roles work together in a team, agile teams often make use of certain team practices, or ceremonies. Stand-up meetings, which are short, daily meetings where team members share their work progress and possible impediments, are used to keep track of the progress of the software (Stray, Sjøberg & Dybå, 2016). Retrospectives are another popular practice. These are meetings where the team members come together to reflect on past work processes; what did we do well, what do we want to keep doing and what do we want to do more of (Gonçalves & Linders, 2013). During a retrospective the team members discuss possible measures that can ultimately improve the sustainability of the team. Furthermore, doing retrospectives frequently is associated with business value in the long run (Gonçalves & Linders, 2013).

On a more general level, there are many approaches to an agile software development process. Two popular ones are Scrum and Kanban (Gren, 2017). Both are frameworks that break down the workload into smaller and more manageable pieces. These chunks of work, or tasks, are often visualized on Scrum or Kanban boards, with a to-do list (backlog), a work-in-progress list and a completion list (Kniberg & Skarin, 2010). In each work interval a number of tasks are solved. The goal with a process like this is to deliver value to the customer in shorter intervals, allowing the customer to get more involved in the process (Dingsøyr et al., 2012). This is in line with Gren's (2017) view on the core practice of an agile team mentioned earlier.

While there are many variants of Scrum and Kanban, and some in between, there are key differences between the two. A considerable difference is how the approaches limit the intervals of work (Kniberg & Skarin, 2010). Kanban limits the amount of work by the size of the backlog. In principle it is possible to have the entire project broken down in the backlog, but usually there is a cap on the number of tasks. Scrum, on the other hand, limits work by time, as tasks are usually carried out in planned intervals, also known as sprints (Kniberg & Skarin, 2010). Scrum also relies on two predescribed roles (Schwaber & Beedle, 2002):

- **Scrum Master***,* who is responsible for overseeing the process during each sprint
- **Product Owner**, who is responsible for the initial planning process and communication with the rest of the organization

Although there are differences between Scrum and Kanban, they both represent a shift away from less total planning towards more interval-based software development (Gren, 2017).

Agile Methodologies and STS

Nerur et al. (2010) argue that today's agile practices in the software development industry align with the principles of STS. They underline that agile methods emphasize an open and collaborative development environment. The developers have the decision-making authority for assigning roles and choosing problem-solving strategies, which is associated with the idea of organizing in self-managing teams in STS. Table

2.2 shows how selected agile principles are rooted in STS, according to Nerur et al. (2010).

Table 2.2: Roots of agile principles (Nerur et al., 2010).

Agile Principle	**Previous Developments**
Emphasis on individuals	Socio-technical systems, e. g. Trist (1981)
Emphasis on work design and work accomplishment	Socio-technical design, e. g. Cherns (1976)
Minimal critical specification	Socio-technical design, e. g. Cherns (1976)

Drawing a parallel from Scrum and Kanban to STS, planning only what is needed resembles Cherns' (1976) second principle of socio-technical design, namely minimal critical specification. There are also other comparisons that support the claim of Nerur et al. (2010), that agile practices are similar to that of STS. For instance, according to Moe et al. (2009), self-managing teams should be responsible for planning and scheduling their own work, as letting the individuals participate will increase their commitment to the team plan. Scrum and Kanban are examples of these bottom-up self-determined work designs, that Parker & Wall (1998) consider a defining feature of autonomous teams. Stand-up meetings and retrospectives are also processes well within the aspect of control and management in the definition given of an autonomous team by Goodman et al. (1988). In other words, as Moe et al. (2009) allude to, the research itself is not new, it has just found a new area of application.

Following the close relation to STS, studies on agile teams have identified many of the same challenges. Moe et al. (2009) distinguish between challenges, or barriers as they call them, on the team level and on the organizational level. The team-level challenges that Moe et al. (2009) identify are individual commitment, failure to learn and individual leadership. Individual commitment is linked to a lack of commitment to the team goals;

they found that team members tended to pursue their own individual goals instead of the team goals. Failure to learn concerns process improvement; even though the team members frequently discussed potential changes, they did not implement them. One reason was that the management did not set aside time for process improvement. Moe et al. (2009) claim that if a team is not given the possibility to improve, it will only experience symbolic self-management, as previously explained by Tata & Prasad (2004).

Additionally, Moe et al. (2009) present the team level challenge of individual leadership. Stray, Moe & Dingsøyr (2011) found that even though agile methods were implemented, critical decisions were in some cases made by the project managers without involving the developers. These findings are supported by Moe et al. (2009). They found that even though the concept of shared leadership was introduced to the teams in their study, team members did not change their individual decision-making processes. This led to difficulties in aligning decisions when team members did not know what others were doing. Important decisions were also made without informing the rest of the team, which led to a low level of trust. In order for this to be successful, Moe, Aurum & Dybå (2012) argue that team members need to identify important decisions they should make together. Otherwise they will not be able to make the shared decisions they are supposed to.

Moving on to the organizational level challenges, Moe et al. (2009) state that the implementation of self-managing teams is difficult if there are barriers on the organizational level. One of these barriers is organizational control. Moe et al. (2009) found that certain forms of detailed control by the management inhibits autonomy, as the whole point is that the teams should control themselves. Boehm & Turner (2005) argue that this is where the project manager comes in; one of the primary roles is to be the barrier between the organization and the team, preventing unnecessary interruptions.

The two other challenges on the organizational level are shared resources and specialist culture (Moe et al., 2009). Shared resources entails that projects fight for resources and the most skilled employees, in an organization rarely building redundancy. In other words, resources are shared across several projects, which threatens the self-management. Specialist culture is a result of organizations supporting and incentivizing being the best at what you do; not creating generalists who can fill each others functions (Moe et al., 2009).

Furthermore, coordinating externally is an issue. According to Boehm & Ross (1989), the primary problem with project coordination is that stakeholders such as users, customers, the development team and the management have to be simultaneously satisfied. This view is supported by Pikkarainen, Haikara, Salo, Abrahamsson & Still (2008), who claim that agile practices do not provide communication mechanisms in situations where many teams are involved in the same development process. Scrum and Kanban are examples of such practices, and as mentioned before, they often make use of visualization tools. However, these tools are normally limited to single teams, and not meant for cross-team communication. As a consequence, according to Pikkarainen et al. (2008), they are not tools for coordinating multiple teams or projects at the same time.

2.3 Defining the Terms

What is Autonomy?

As explained by Moe et al. (2009), teams face challenges on a team level and on an organizational level. Autonomous teams need to make team members commit to shared goals, they need to make shared decisions on a team level and they need to coordinate and communicate with their environment. Based on the team’s ability to achieve these feats, Moe, Dingsøyr & Dybå (2008) present three different levels of autonomy: *individual autonomy, internal autonomy* and *external autonomy*.

Firstly, Van Mierlo, Rutte, Vermunt, Kompier & Doorewaard (2006) and Langfred (2000) in Moe et al. (2008) refer to *individual autonomy* as the individual's freedom and discretion when carrying out an assigned task. According to Langfred (2000), individuals with a high level of control of the nature and pace of their work are defined as having high individual autonomy. In contrast, when the task interdependence is high, the individuals need to coordinate their efforts, resulting in lower individual autonomy (Langfred, 2005). Further, some suggest that increased individual autonomy augments the intrinsic motivation of a team member (Stewart & Manz, 1995).

Internal autonomy refers to in what degree all team members jointly share the decision-making authority (Moe et al., 2008). The internal autonomy is regarded as low given a centralized decision-making structure, that is if for example one person is making all the decisions. Given a decentralized decision-making structure, where team members make decisions for their own work individually and independently of each other, the internal autonomy is also regarded as low (Moe et al., 2008).

Lastly, Hoegl & Parboteeah (2006) define *external autonomy* as the influence of managers and other team external individuals on the team's activities. External autonomy measures to what degree the team needs to include higher level managers or other individuals outside the team in their decision-making. The influence of management can (deliberately or not) limit the autonomy of the team, prompting certain decisions for strategy, processes, goals or allocation of resources. If external actors only exercise limited control over the team, the external autonomy is high. The external actors may be departments, teams or individuals in the rest of the organization, or customers and their teams (Hoegl & Parboteeah, 2006).

According to STS proponents, the social component of autonomous teams is closely linked to the psychological needs of the employees (Manz & Sims, 1982; Trist, 1981). Hackman & Oldham's (1980) view on some of these needs, e. g. responsibility for outcomes, were presented earlier. Hackman & Oldham (1980) state that autonomy is an important characteristic in fostering a personal feeling of responsibility. They define autonomy as "*the degree to which the job provides substantial freedom, independence, and discretion to the individual in scheduling the work and in determining the procedures to be used in carrying it out*" (Hackman & Oldham, 1980, p. 79). This definition is very similar to individual autonomy as presented in Moe et al. (2008). Subsequently, for the purpose of this thesis, when we refer to individual autonomy by Moe et al. (2008), Hackman & Oldham's (1980) definition of autonomy is covered.

Research Question

To recall, autonomous teams face several challenges. Among these are shared decision-making (Moe et al., 2012; Wageman & Fisher, 2014), the need for new skills and knowledge (Langfred, 2007), maintaining a flow of information (Lawler, 1992) and achieving genuine self-management (Tata & Prasad, 2004). For agile teams, Moe et al. (2009) point to challenges on a team level and on an organizational level. As mentioned, some of these challenges resemble those presented by proponents of STS.

In this thesis we will refer to the term *challenge* as a matter that impedes the team's processes. We will identify the challenges of the teams in this study using an inductive approach. They are entirely based on the findings of our multiple case study. After identifying the challenges, we will look at how they relate to autonomy, discussing them in light of STS, agile and other relevant theory. In order to highlight various aspects of being autonomous, we will refer to the different definitions of autonomy presented in this section.

Hence, our RQ is:

What are the challenges for autonomous development teams, and how do they relate to autonomy?

It is important to note that by teams in this thesis, we refer to "*a small number of people with complementary skills, who are committed to a common purpose, performance goals and approach, for which they hold themselves mutually accountable*" (Katzenbach & Smith, 1993, p. 112). Some teams face more complex tasks than others. Ratliff, Beckstead & Hanks (1999) define the term *problem-solving teams*. These are teams that combine a variety of skills and knowledge to solve problems with unclear structure and boundaries. Task-solving often entails exploring new areas and producing something unique; not doing standard work to solve routine tasks (Ratliff et al., 1999). The teams we study in this thesis develop software, and arguably fall into the category of problem-solving teams. Given the nature of their work, we more specifically consider them to be *autonomous development teams*. When we refer to teams in this thesis, we refer to this category of teams.

3 Methods

This chapter outlines the methodology for this thesis, as shown in Figure 3.1. First, the research context is presented. Then the research design, followed by the research methods for data collection and analysis, are described. We end the chapter with a discussion and evaluation of our methods.

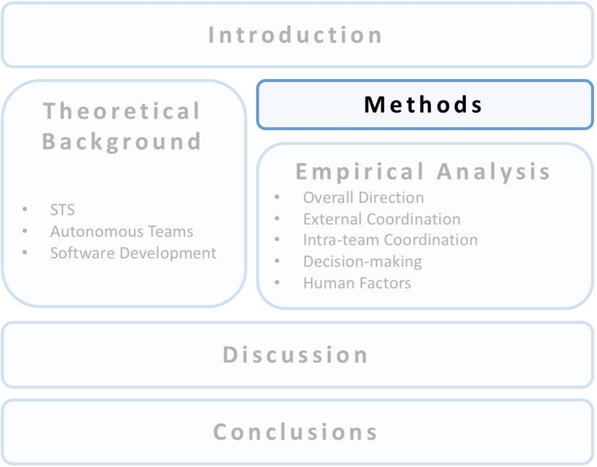


Figure 3.1: Structure of the methods.

3.1 Research Context

This master's thesis is a part of the A-team project at SINTEF. The objective of the project is to provide knowledge on how to achieve high performance with autonomous development teams in complex environments. Four companies from two areas of Norwegian industry (finance and consulting) takes part in the project. Autonomous teams in these fields are ideal for team studies, because they operate with short time spans from idea to solution, making it possible to follow a team through several phases in a short period of time.

These teams form the foundation for the empirical research in this thesis. Three of the four companies are included in this study, and are presented in Chapter 4. We have interviewed several team members from a number of the companies' teams, which are all aiming to become autonomous high performance teams. Some of the teams are also cross-functional.

3.2 Research Design

Our thesis applies a *qualitative research strategy* and a *multiple case study design* to provide a framework for collection and analysis of data (Bryman, 2016). The cases in our study are considered *common* or *representative cases*, where the objective is to capture the circumstances of a commonplace situation (Yin, 2014). This means that the cases are not chosen because of their unusual nature, but rather because they are considered to provide a proper context to answer our RQ concerning autonomous development teams. Our goal is not to compare the case companies, but rather to build a wide understanding of the challenges and their relation to autonomy.

Generally, a qualitative research strategy orients towards an *inductive approach*, emphasizing words rather than quantification of data (Bryman, 2016). We have chosen an inductive approach to identify challenges for autonomous teams in the case companies and explore how they relate to autonomy. Thereby, we motivate and form the research by our empirical findings. The empirical analysis (presented in Chapter 5) forms the basis for our discussions, and relevant theory is used to enlighten our findings. Our ontological position is *constructionism*, which emphasizes that social reality is a result of the interactions between individuals. The epistemological position of this research is *interpretivism*, as we aim to examine the social world through the interpretation of its participants (Bryman, 2016).

3.3 Research Method

The choice of a qualitative research strategy has implications for our approach of collecting data. The research method applied in this thesis is considered a well-known approach in case study research; *semi-structured interviews* (Bryman, 2016; Yin, 2014). Before initiating the data collection we had to determine the sample of informants.

3.3.1 Sampling of Informants

The sampling of informants was conducted through *purposive sampling*, which is considered a central principle for selecting cases and individuals in qualitative research. The goal of purposive sampling is to sample informants strategically, so that those sampled are relevant for the posed RQ (Bryman, 2016). In our study, we have included informants from three of the companies already participating in the ongoing A-team project at SINTEF. Our contacts at SINTEF helped us get in touch with informants in the different companies. This sampling approach was considered convenient, because it gave us easy access to informants within the case companies.

In order to answer our RQ about challenges for autonomous teams, the sample consists of team members and higher level managers from the three case companies. Our data collection started out with conducting semi-structured interviews of team members holding formal leadership responsibilities, as they were easily accessible and considered helpful in the identification of challenges. We also interviewed some managers from higher levels of the organizations. During the process, we found interesting dynamics related to our RQ within certain teams. Hence, we chose to interview more members of these teams to gain more insight. This forms our two categories of informants; informants with, and without formal leadership responsibilities. By including informants from both categories in our research, we aim to provide a more holistic insight into what challenges that are present in the teams, how they are approached and how they relate to the autonomy. Chapter 4 presents a more thorough description of the case companies included in our research.

3.3.2 Semi-structured Interviews

Semi-structured interviews were used to gather information. Semi-structured interviews are flexible and can provide important insights into how the research participants view the world (Bryman, 2016). Additionally, interviewees can be interviewed on more than one occasion, which is an option we made use of. This type of interview is also preferable when more than one person is carrying out the fieldwork, as it ensures some comparability of interviewing styles (Bryman, 2016). Further, the nature of the semi-structured interview allows the informants to elaborate on topics that they consider relevant. By allowing digressions, topics that has not been thought of ahead of the interview, but that seem important to the informant, can be uncovered (Tjora, 2017). An example of such a topic that emerged during our interviews is how the teams consist of specialists and not generalists.

Creating the Interview Guides

Before conducting the interviews, we created two different interview guides; one for informants holding leadership responsibilities (Appendix A), and the other for informants without any formal leadership authority (Appendix B). According to Tjora (2017), the semi-structured interview is typically divided into three phases: warm up, reflection and closure. The three phases require different kinds of questions and demand varying degrees of reflection from the informants. Our interview guides follow the three phases suggested by Tjora (2017), and have a relatively broad focus. The interviews started out with a warm-up phase, consisting of concrete questions that were presumptively easy to answer. These questions were mainly about practical information and generalities concerning their work situations. The next phase was considered “the core” of the interview, and the questions in this phase required more reflected and in-depth answers. Here, we made inquiries about more complex topics, such as challenges the informants deal with at work. Next, the informants were asked to elaborate on how they would cope with different scenarios of conflict. The closing phase gave us the opportunity to sum up the interview, and the informants were encouraged to share thoughts about topics that had not already been addressed.

Before interviewing the actual informants, we tested the interview guide for informants holding leadership responsibilities on two learning assistants from the course “Experts in Teamwork” at NTNU. The learning assistants hold the role as facilitators to the student teams in the course, and were therefore considered as appropriate pilot interviewees for the purpose of our research. Through these pilot interviews we received valuable feedback which we used to further develop our interview guides. Based on their suggestions, we made adjustments to the wording of several questions to make them more intuitive and easier to understand. Additionally, since the interview guides consisted of a couple of constructed scenarios we wanted the interviewees to elaborate on how they would approach, it was important for us to get feedback on them. This appeared to be a difficult exercise for our pilot interviewees, and with their feedback we were able to improve the scenarios. Finally, all our adjustments along with the final interview guides were approved by our supervisors at SINTEF and NTNU.

Conducting the Interviews

The interviews were conducted as semi-structured conversations in line with the interview guides. Since informants were located in different cities around Norway, Skype was used for some of the interviews. Bryman (2016) states that the convenience of being interviewed via Skype can make it easier to get informants to participate in a study. It is flexible, time-saving and therefore more convenient for the interviewee to take part. However, we experienced some technical difficulties in terms of unstable internet connections and bad sound quality, which led to some poor recordings. It was also disturbing for the conversational flow. Because of these inconveniences, we conducted as many interviews as possible face-to-face. These interviews were held at the informants’ workplaces. According to

Tjora (2017), it is common practice to conduct a semi-structured interview at a location where the informant feels confident, as it contributes to a more relaxed atmosphere.

We carried out the interviews in February, March and April of 2018. In each of the interviews at least one of the three researchers were present. Our co-supervisor from SINTEF and two other researchers in the A-team project also participated in some of the interviews. Several of the informants considered the SINTEF researchers' presence as essential in order to participate in the interviews, because the supervisors, more so than us, could provide direct feedback on issues that concerned them. Additionally, the researchers from SINTEF had interacted with many of the informants on earlier occasions. We think this contributed to more honest and reflective answers from the informants, as many of them already knew one of the interviewers. The researchers from SINTEF are also experienced and have extensive training in conducting interviews, which gave us something to reach for when conducting the other interviews ourselves.

The quality of an interview depends on the trust built between the informant and the researcher. To ensure that the informant becomes familiar with the situation, a semi-structured interview should have a certain duration (Tjora, 2017). When planning the interviews, we aimed at a duration of 45 to 60 minutes per interview for the informants with formal leadership responsibilities. The average length of these interviews was 50 minutes. For the informants without formal leadership responsibilities, the aim was 30 to 40 minutes. The resulting average length was 36 minutes.

Table 3.1 shows an overview of the interviews with informants with formal leadership responsibilities.

Table 3.1: Informants with formal leadership responsibilities.

Role	**Case Company**	**Setting**	**Number of Interviews**
Business Representative	A	Skype	1
Product Owner	A	In person	1
Product Owner	A	Skype	1
Project Manager	A	Skype	1
Team Lead	A	Skype	2
Business Representative	B	Skype	1
Line Manager	B	Skype	1
Tech Lead	B	Skype	1
Tech Lead	B	Skype	1
Tech Lead	B	Skype and in person	2
Line Manager	C	In person	1
Project Manager	C	In person	1
Team Lead	C	In person	1
Team Lead	C	In person	1
Team Lead	C	In person	1
Team Lead (and Tech Lead)	C	In person	2
Total number of interviews in this category			19

Table 3.2 shows an overview of the interviews with team members without formal leadership responsibilities.

Table 3.2: Informants without formal leadership responsibilities.

Role	**Case Company**	**Setting**	**Number of Interviews**
Domain Architect	A	Skype	1
Developer	B	In person	1
Developer	B	In person	1
Developer	B	In person	1
Developer	C	In person	1
Developer	C	In person	1
Developer	C	In person	1
Total number of interviews in this category			7

3.4 Analyzing the Data

We analyzed the gathered data in a Thematic Analysis (TA). According to Braun & Clarke (2013), TA is a suitable method for data analysis in qualitative research, and has few restrictions regarding research methods and theoretical positions. This makes the method flexible and useful for any kind of RQ or data material. Also, TA is recommended for those new to qualitative research, in particular for student projects (Braun & Clarke, 2013).

As described in Section 3.3.1, the collection of data was done in two rounds. First, we interviewed informants holding leadership responsibilities. The findings from these interviews formed the foundation for the second round of follow-up interviews and interviews other team members. The steps of the data collection, coding and thematization is presented in Figure 3.2, where the arrow to the left represents going back for the second round of data collection.

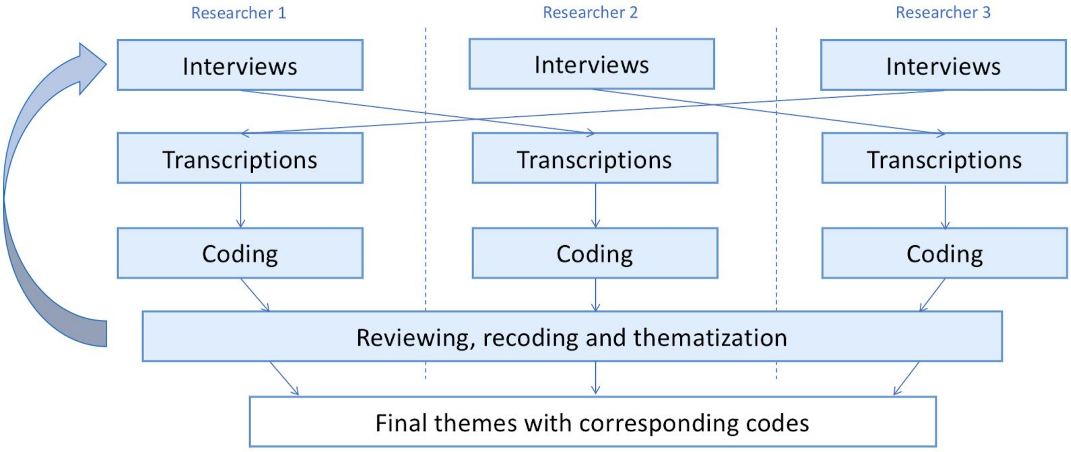


Figure 3.2: The process of gathering, coding and analyzing empirical data.

Interviews and Transcriptions

The first step was to conduct the first round of *interviews*. This was followed by *transcription* of the recordings, which entails converting the audio files into written text (Braun & Clarke, 2013). We used Express Scribe as a tool to help us transcribe the audio recordings. The software is designed to simplify the process of transcribing, and reduced the turnaround time. The transcription process was followed by a read through in order to take note of items of particular interest. To ensure that all three researchers achieved a full overview of all the collected data, the interviews were, if possible, transcribed by someone that had not participated in the interview.

Coding

The next step in the process was to *code* the transcriptions. A code is a word or brief phrase capturing the essence of why the data under the code is useful (Braun & Clarke, 2013). The coding was carried out using a software called NVivo; a digital software tool for analysis of qualitative data. We applied the approach of complete coding, which means coding all the data relevant for the RQ. The process is systematic, coding all data that may potentially be relevant (Braun & Clarke, 2013). The interviews were divided into three arbitrary groups, and coded separately by the three researchers. However, Braun & Clarke (2013) state that when several researchers do coding, there is likely that differences will occur. To reduce the chance of these differences becoming problematic, we coded a few interviews together to gain a common understanding on how to code. Examples of codes that were formulated are: ‘shared direction’, ‘coordination ceremonies’, ‘role descriptions’ and ‘responsible for decisions’.

Reviewing, Recoding and Thematization

Then we *reviewed the codes,* followed by a process of *recoding* and *thematization*. The reviewing was done by gathering the codes made by the different researchers, with the intention of looking for similarities and differences. This was important in order to discover any immediate inconsistencies and ensure all relevant information was coded. Next, samples of transcriptions were switched between the researchers, and recoded by a different researcher. When the codes diverged, they were retraced and updated accordingly.

After the recoding, we looked for patterns across the data sets. The broader patterns are called *themes* and comprises several codes. A theme has a central organizing concept telling something about the content of the data in relation to the RQ (Braun & Clarke, 2013). We then formed an idea of which themes that could possibly be a part of our final analysis. These tentative themes were kept in mind during our second round of data collection to gather more applicable information.

Second Round of Data Collection

After completing the four steps of the process, a second round of data collection was conducted. This is illustrated by the upward pointing arrow on the left side in Figure 3.2. This time, the main sample of informants were team members without any leadership responsibilities. In addition, we conducted second time interviews with some informants from the first round. The interviews were processed through the same four steps as described.

Final thematization

After transcribing, coding, reviewing and recoding the second round of data collection, we started a final thematization. We tried to begin this process with open minds, in an attempt not to be affected by the tentative themes. The focus was on finding themes that could help us identify challenges, and how they relate to autonomy, as to answer our RQ. During the thematization, we had to let go of codes we found interesting, but that was not a part of our scope. Some of the findings we could not take further are included in the suggestions for further research in Section 7.3.

We started the final thematization by looking for patterns across all the data gathered. This was a very important activity in our research process. Because of our inductive research approach, the themes identified from the data material form the basis and structure of our further analysis. We identified five final themes: Overall Direction, External Coordination, Intra-team Coordination, Decision-making and Human Factors. These themes highlight important aspects of the challenges in autonomous teams. A presentation of the themes with corresponding descriptions are presented in Table 5.1, followed by the full analysis of each of them.

In the empirical analysis presented in Chapter 5, excerpts from the data are chosen to illustrate and exemplify the aspects of each theme. In line with recommendations from Braun & Clarke (2013), the excerpts are selected carefully, as they are the only parts of the data the reader can see.

3.5 Evaluating the Research Methods

This section assesses the quality and ethical issues of this research. First, the quality of the research will be evaluated based on principles provided by Lincoln & Guba (1985) and Guba & Lincoln (1994). Then, we discuss the ethical aspects of our research.

3.5.1 Quality of the Research

Bryman (2016) refers to Lincoln & Guba (1985) and Guba & Lincoln (1994), who assess the quality of qualitative research in an alternative way to reliability and validity. They propose two quality criteria for a qualitative study, namely *trustworthiness* and *authenticity*. We assess the quality of our research based on these criteria.

Trustworthiness

Trustworthiness consists of four criteria: *credibility*, *transferability*, *dependability* and *confirmability*. Firstly, credibility entails that the findings are validated by members of the social world, and that the research has been carried out in accordance to principles of good practice (Lincoln & Guba, 1985). Our sample consisted of two categories of informants from the three case companies, therein informants with and without formal leadership responsibilities. The inclusion of the different categories of informants was done to provide a more holistic insight into what the challenges in the teams are and how they are approached. Further, to ensure high quality of the interview guides, they were tested ahead of the data collection and approved by our experienced supervisors at SINTEF and NTNU. Further, all informants consented to being recorded and later transcribed. The comprehensive transcription process was conducted to make sure that meanings and context did not get lost during our analysis. The transcribed interviews were also stored in original form, so that no data was lost.

Secondly, transferability refers to which aspects of the research that hold for other contexts. An important term here is thick description: a very detailed description that leaves what is holdable in other circumstances up to the reader (Lincoln & Guba, 1985). We have included a description of the three case companies in Chapter 4 in order to provide, as far as it is possible, a complete impression of the social context of the informants. However, when evaluating rich data against the anonymity of the informants and the case companies, we have prioritized to ensure the anonymity of our research participants. This might be at the expense of the transferability of our study. Still, by including case companies from both the finance and the consulting sector, we think several findings can be applicable to autonomous development teams operating under similar conditions as the teams in our sample.

Thirdly, dependability is parallel to reliability, and Lincoln & Guba (1985) suggest that the researcher should keep complete records of all phases of the research that are accessible by peers. To ensure dependability of our study, we have kept record of the different phases of our research in this chapter (Chapter 3). Further, we have kept a close dialogue with our supervisors at SINTEF and NTNU, to make sure we carry out our research according to good research practice.

Lastly, confirmability concerns acting in good faith; although complete objectivity is unattainable, the researchers should refrain from letting personal values and interests interfere in the research (Lincoln & Guba, 1985). We have structured our analysis with the intention of not letting our personal interests and opinions interfere with our findings. By presenting the empirical analysis in Chapter 5, separate from our discussions of the empirical findings in Chapter 6, we attempt to show a logical structure of reasoning for the reader of this thesis.

Authenticity

Authenticity concerns the political and societal side of the research (Lincoln & Guba, 1985). We believe that our research will contribute to a better understanding of challenges that exist in autonomous team, and how the teams' approach to these relate to the autonomy. Our master's thesis is written in cooperation with SINTEF and their A-team project, and we hope our research will be a meaningful contribution to their final delivery, and to the case companies. We also hope this thesis will inspire further research suggested in Section 7.3, or research on other related topics.

3.5.2 Ethical Discussion

During our study, we emphasized the use of ethical research methods. As a result, we made a number of ethical considerations. Before gathering data from the informants, we sent an application to the Data Protection Official at the Norwegian Centre for Research Data, which was approved (Appendix D). Further, in order to ensure that the informants understood the implications of participating in our research, a consent form (Appendix E) was signed by the informants in advance of the interviews. The consent form explicitly states that all information will be anonymized, and that the informants at any time of the study can withdraw their participation. Before starting the interviews, we asked for permission to audio record the interviews. All of the informants agreed to be recorded, and the audio files were kept in a locked digital folder under SINTEF's security restrictions. The recordings will be deleted in the end of the research project.

Further, we have chosen to refer to all informants as “he” when presenting and discussing the data material, regardless of the gender of the informant. We have also changed some of the role titles of our informants (presented in Table 3.1 and 3.2). Specific and identifiable titles are replaced by more general and overarching titles. These measures were made in order to ensure anonymity of the research participants and case companies.

4 Presenting the Case Companies

This chapter presents the three case companies in this thesis. An overview of the interviews from the different case companies were presented in Chapter 3. The case descriptions are kept brief, including only the information considered necessary to follow our analysis. A summary of the main characteristics is shown in Table 4.1.

Table 4.1: Characteristics of the case companies.

	Company A	**Company B**	**Company C**
Team Organization	Company internal teams	Company internal teams	Consulting teams for customers
Type of Team	Cross-functional development teams	Cross-functional development teams	Technical development teams
Leadership Roles within the Teams	Product Owner and Business Representative	Tech Lead, Product Owner and Business Representatives	Team Lead and Tech Lead
When Teams Were Introduced	Has been the practice for about a year	Has been the practice for about a year	Has been the practice for several years

4.1 Company A

Case company A has introduced cross-functional teams within and across departments as a new practice. The teams combine business and ICT development. These are departments that in the past have been kept separate. Traditionally, the relationship has been similar to a buyer supplier relationship; the business side initiates and ICT tries to answer their requirements. Although the teams are newly organized, the organizational structure remains the same overall, with line configurations. Figure 4.1 shows how the teams put together workers from the different departments, and how the teams create horizontal dependencies.

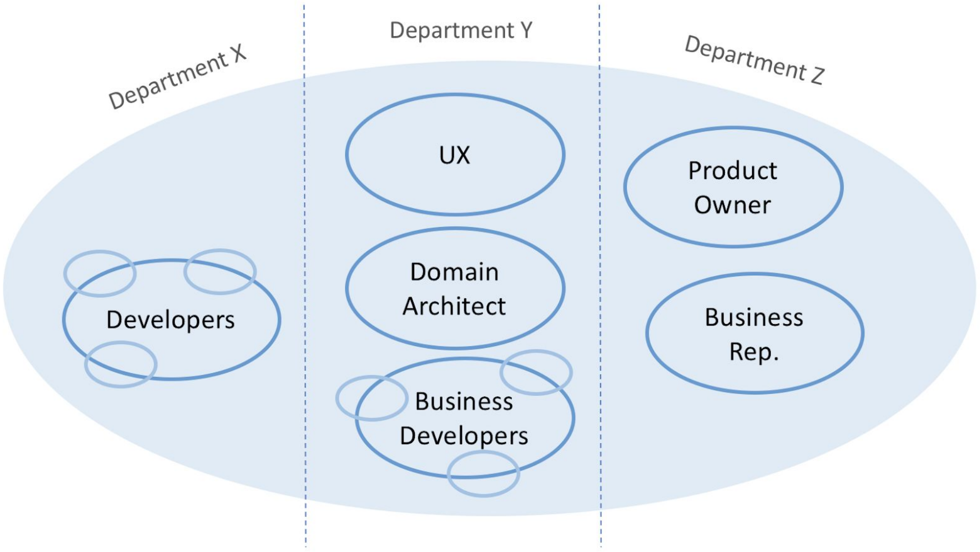


Figure 4.1: Team structure in company A.

Table 4.2 presents the team roles we have found within company A. The responsibilities associated with each role are based on how the informants explain their own roles.

Table 4.2: Roles in company A.

Role	**Responsibility**
Domain Architect	A tech role responsible for the breadth of the system and its functions.
Project Manager	A leader responsible for larger groups of teams, or organizational programs.
Product Owner	A leader responsible for the solutions the team produces. Usually the contact point for external surroundings.
Developer	Responsible for writing the code, often front-end or back-end.
Business Representative	Responsible for ensuring that the delivery takes place and is satisfactory. Also has some internal coordination responsibility.
Business Developer	Develops the business parts of the solution, what the solution is actually made to do.
UX Designer	Ensures that the solution is user friendly and what the users want.

4.2 Company B

Case company B has also introduced cross-functional teams within and across departments, combining business and ICT development. The company implemented a restructuring of the organization about a year ago to better adapt to this new way of working. The transition to working in autonomous teams is still an ongoing process, where the development teams are changing their work processes. Figure 4.2 shows how the team boundary may not always be clear, and how the team coordinates with higher level managers outside the team.

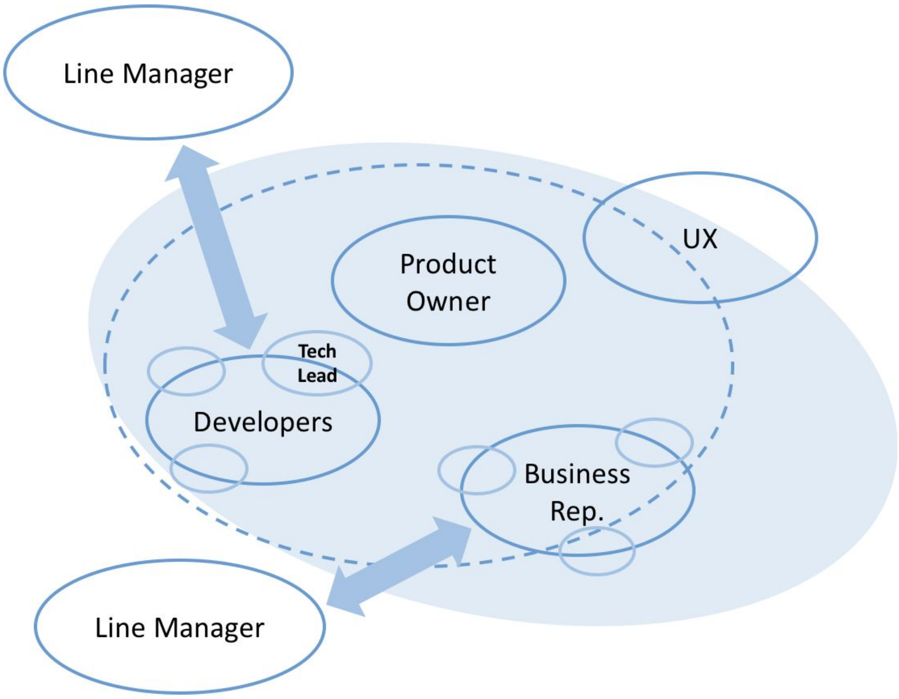


Figure 4.2: Team structure in company B.

In company B, the business side initiates deliveries and ICT tries to answer their requirements. Initiation of teams, specification of the team's tasks and who joins the teams is decided by higher level management from different departments. Table 4.3 presents the roles we observe in company B. The responsibilities associated with each role are given as the informants explain them, or as they are defined by the organization.

Table 4.3: Roles in company B.

Role	Responsibility
Line Manager	An external leadership role responsible for follow-up of the teams. Holds staff responsibility.
Tech Lead	A new role formed during the reorganization. Responsible for distributing, detailing and prioritizing the tasks of the team together with the Business Representative, as well as the team's progression and deliveries. Also responsible for coordination with external surroundings, scope definition, motivating the team and building culture.
Business Representative	Two different descriptions: 1. Responsible for the team's financial results and the financial content of the team's deliveries (this role definition has not been changed after the reorganization). 2. Defines what the team is going to develop and initiates new projects or tasks for a team.
Product Owner	Responsible for the prioritization of tasks, together with the Business Representative.
Developer	Writing different parts of the code. Divided into front-end and back-end developers.
UX Designer	Ensures that the solution is user friendly.

4.3 Company C

Case company C is a consulting company. This means that an external customer hires a team of consultants from the company to develop a product for them. The company has had a team-based structure for several years. There are line managers responsible for initiating projects from customers, and putting together the right team of consultants to develop the desired solution. In addition, there are groups and forums across the teams to ensure individual development of the consultants, coordination of knowledge and discussion on role specific challenges. Compared to company A and B, these teams are less cross-functional and more technical. A general model of a consulting team in case company C is shown in Figure 4.3.

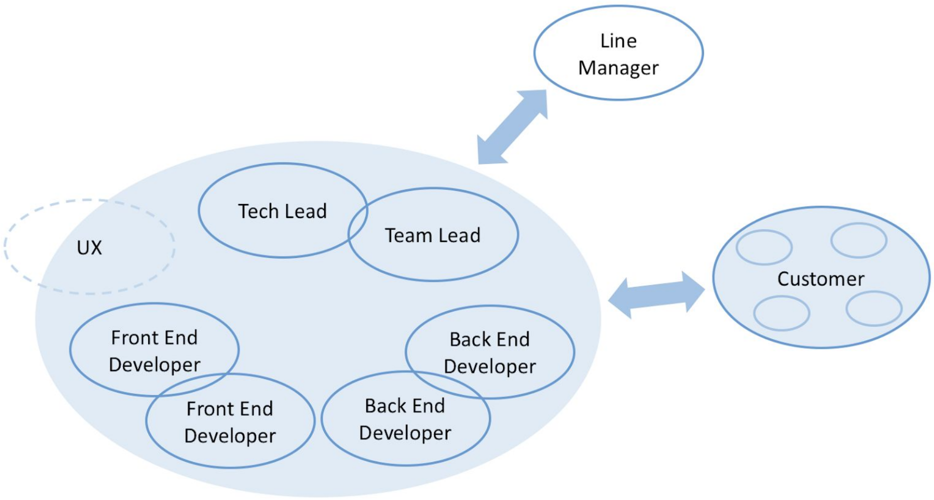


Figure 4.3: Team structure in company C.

The customer serves as a product owner, and has one or several contact persons for the team, setting requirements for the solution. The characteristics of the teams are decided and limited by the customer's priorities, such as product criteria and budgets. Table 4.4 presents team roles in company C. Their responsibilities are given as explained by the informants.

Table 4.4: Roles in company C.

Role	Responsibility
Line Manager	Takes care of the mercantile part of the work, such as initiating new projects with customers, counting hours and ensuring the right amount and type of resources are in the teams.
Project Manager	For larger projects consisting of several teams, there might be a project manager responsible for coordination among teams and with the customer.
Customer	Orders the product and coordinates closely with the team to ensure deliveries as desired.
Team Lead	Clears away obstacles, makes sure everyone has things to do, answers questions, manages the dialogue with external surroundings, provides relevant information, shields the team from things they do not need to know, motivates and keeps overview.
Tech Lead	Has a architectural responsibility for the product, and coordinates the other developers.
Developer	Writes different parts of the code. Divided into front-end developers and back-end developers.
UX Designer	Ensures a user friendly solution.

5 Empirical Analysis

Chapter 4 presents the case companies. This chapter presents the main results from the analysis of the empirical data gathered in this thesis, as shown in Figure 5.1. The data derives from semi-structured interviews, and are analyzed thematically (TA) as presented in Chapter 3. Five themes evolved from the analysis.

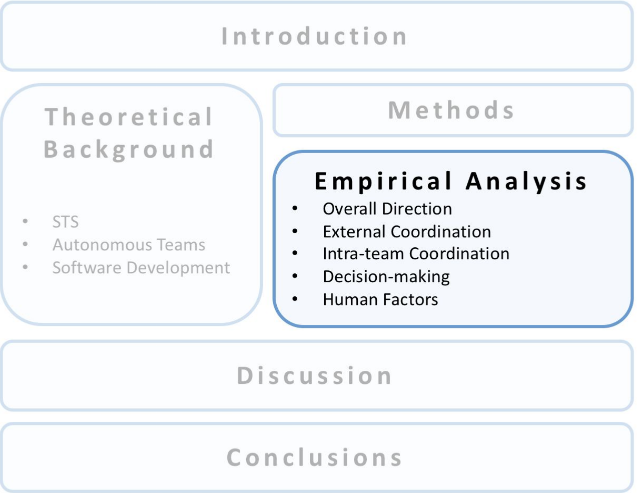


Figure 5.1: Structure of the empirical analysis.

First, we will shortly present a general bewilderment regarding the term *autonomy*, to serve as a backdrop for the empirical analysis. A developer from company C exemplifies how the meaning of the term is not well known nor easy to understand, as he had to google it before he came to the interview. However, he is not sure if he captured the essence of it. Several developers from company B stress necessary knowledge and resources to implement their activities as the most important features of autonomy. A team lead from company C adds to this, reflecting on how everyone talks about autonomy without a common definition of it. This confusion is further explained by a product owner from company A, questioning whether higher level managers know what autonomy is really about.

At the same time, the informants speak warmly of autonomy. A business representative from company B considers himself a supporter of autonomous teams, and a line manager in the same company talks passionately about this way of working. A developer from company C agrees, highlighting the freedom in how to develop the solution and working closely together, as features he appreciates. Furthermore, on the question of whether the informants see their teams as autonomous, the general answers are “yes” or “almost”. The highlighted differences in interpretations among the informants, as well as the view on their teams being autonomous to certain degree, illustrates the difficulty of autonomy.

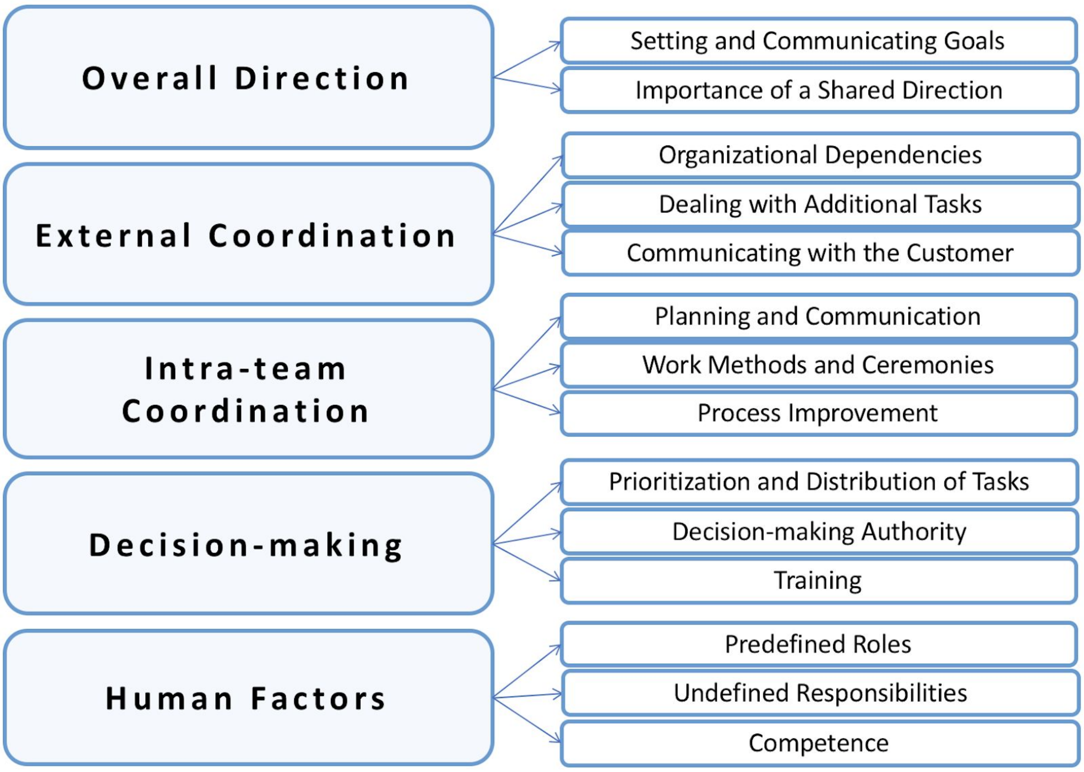

Overall Direction
Setting and Communicating Goals
Importance of a Shared Direction
External Coordination
Organizational Dependencies
Dealing with Additional Tasks
Communicating with the Customer
Intra-team Coordination
Planning and Communication
Work Methods and Ceremonies
Process Improvement
Decision-making
Prioritization and Distribution of Tasks
Decision-making Authority
Training
Human Factors
Predefined Roles
Undefined Responsibilities
Competence

Figure 5.2: The five themes from the TA.

Moving on, the five empirical themes from the TA are shown in Figure 5.2. These themes are seen as important to describe the challenges for autonomous development teams, and how they relate to autonomy. To provide an overview, the themes from the TA will first be presented briefly in Table 5.1, as suggested by Braun & Clarke (2013). Then, the more comprehensive analysis of the empirical data corresponding to each theme will follow.

Table 5.1: Presenting the themes.

Theme	Description
Overall Direction	This theme elaborates on how the teams in the different case companies set an overall direction. Two sub-themes are identified and likely to affect this manner. The first concerns how goals for the teams are set, what they entail and how they are communicated to the team. The second sub-theme examines the emphasis on creating a shared direction among team members.
External Coordination	External coordination entails how teams coordinate with their environment. This includes both the rest of the organization and potential customers. The theme is split into three sub-themes. The first examines how organizational dependencies depict how a team relates to other teams and components. Next, dealing with additional tasks concerns what a team does when it receives tasks from the external surroundings. Finally, the third sub-theme explores how teams communicate and coordinate with an external customer.
Intra-team Coordination	Intra-team coordination concerns what work processes the teams have, and what they do to improve them. This includes the level of planning, the use of coordination mechanisms such as stand-up meetings, and improvement and feedback processes such as retrospectives. There are three sub-themes: planning and communication, methods and ceremonies, and process improvement.
Decision-making	This theme explores the decision-making processes in the teams. There are different ways to distribute and prioritize tasks, and various views among the teams on how involved the members, higher level managers and potential customers should be in these processes. From this, two of the three sub-themes are identified as prioritization and distribution of tasks, and decision-making authority. The third sub-theme comprises training in new roles and shared decision-making.
Human Factors	Human factors explores how the teams are formed. This theme is compiled by three sub-themes. The first elaborates on the predefined roles within the teams. The second looks into how undefined responsibilities. The third sub-theme explores the competence and skills of the team members.

5.1 Overall Direction

Setting and Communicating Goals

A developer from company B states that goal setting is mainly done by the business department in the company, and that it is not really the team that formulates the goals. A tech lead supports this, adding that after the goals are set, they are given to the team. However, a line manager from the same company expresses that the team should be involved in setting team goals, and considers this an important feature of autonomous teams. In company C, there are different practices depending on the type of goals and the teams. A project manager states that he typically sets the team's weekly goals. For another team in company C, the team lead clarifies how goal setting processes depend on the size and scope of the goals:

> *(...) the goals are set by those conducting them. But when there are bigger goals concerning deliveries, the whole team takes part.*
>
> *Team Lead, C*

Also, team goals appear to be equivalent to delivery deadlines. A developer from company B expresses that the goals mainly have one focus, and that they often are related to deliveries:

> *(...) lately our only goal is related to deliveries. It is about finishing something at a given time.*
>
> *Developer, B*

Furthermore, the goals and deadlines set by higher level managers do not always appear achievable to the team members. According to a developer from company B, the deadlines are not very rigid:

> *I think they set the deadline with the intention of giving us something to work towards. And then we just have to see if we reach the deadline, or if we have to postpone the date or reduce the scope of our work. In my opinion, the deadlines do not always make sense.*
>
> *Developer, B*

When questioning whether or not the team has any other goals than specific deadlines, a developer from company B answers that his team does not have that many goals, but that one of the departments recently sat some new ones. However, the informant is not exactly sure what the new goals are.

A business representative from company B explains how the goals are mainly communicated through oral communication. The informant further explains that the team members should always have an understanding of the goals. They try to get a more visual tool in place, so that the team can get an impression of the situation with regards to the goal. However, they do not arrange any frequent meetings to talk about the direction of their work. The business representative says that they still communicate a lot with the developers about the direction of their work, and that the impression is that the developers have a holistic picture of what they are doing.

The Importance of a Shared Direction

Several informants from the different case companies emphasize the importance of creating a shared direction for the team. A business representative from company A emphasizes that finding common ground and clarity of goals are necessities when implementing autonomous teams:

> *(...) try to establish what we as a team have to accomplish. Lay that fundament, ensure that people know the goal and vision of the whole (...) So we spent a lot of time, many sessions, finding common ground.*
>
> *Business Representative, A*

Building on this, a team lead from company C states that a team needs someone that possesses an overview of the team's assignment. This is supported by a developer in the same company who expresses that a team needs someone with experience to help pull the team in the right direction. A developer from company B argues in line with the team lead, expressing that experienced developers are important for achieving an effective team. The informant further states that it is very important that everyone in the team has an impression of the overall direction of the company, in order to become an effective team.

The larger picture is also emphasized by a project manager from company C. If the team does not have an impression of the overall direction of its work, it can be difficult to align the team's tasks. He comments on how the relation to the final delivery may inhibit the team members' work:

> *The problem was that the team did not have any relation to the service delivered to the customer.*
>
> *Project Manager, C*

The importance of getting the team to pull in the same direction is a widespread view among the informants. However, how to achieve an overall direction is unequally emphasized by the team members. While a business representative from company B states that meetings regarding evaluation of goals are more ad hoc, others give priority to creating an arena for the team members to involve in setting the direction.

5.2 External Coordination

Organizational Dependencies

Although the autonomous teams studied have a broad spectrum of competence, they are still dependent on other teams, projects, departments and/or systems within the organization, and vice versa. This is exemplified by a domain architect from company A, who explains how his team has to coordinate with other teams:

> *We do not live in our own world (...) one has to coordinate with other teams who share components with your team.*
>
> *Domain Architect, A*

Adding to this, a developer from company B states that the team frequently needs to clarify different issues with the business department. Unclear specifications and need for confirmation are examples of these issues, where the team has to make contact before moving on. Further, a tech lead from the same company states that the team needs to clarify issues frequently with other departments. He elaborates how this need probably stems from not synchronizing each other's work flow.

A team lead from company C also comments on collaboration, and on how different parts of the organization approach the same problem in various ways:

> *Collaborating with others across the organization does not always work well, because they have their way and we have our way* [of working].
>
> *Team lead, C*

Another common issue is that teams have to wait for other resources to finish their job before they can move on. In the case of company C, which is a consulting company, the customer might have multiple teams from multiple companies working on the same end product.

Furthermore, a team lead from company C expresses that the team has many external dependencies, which render them unable to move on even if they have finished their own work. They have to wait for the others to finish.

> *We have external dependencies. We had some cases when integrating with systems made by external teams, and they were either not ready or it was not documented well enough, or we could not even access it (...) And you always have a lot of cases going on that you cannot finish because you have to wait for others.*
>
> *Team Lead, C*

Dealing with Additional Tasks

Several informants state that additional tasks delay the teams in doing their initial work. One of the interviewees from company B says that the team receives a stream of unrelated tasks, which forces the team away from the tasks they are supposed to do. A developer in the same company states that examples of such tasks can be errors in previously developed products that need to be corrected. This is similar to company C, where a team lead states that these unrelated tasks potentially postpone entire sprints (the team is using Scrum). Further, another team lead from company C states how he spends most of his time in meetings and doing administrative work for the team, which is very time consuming.

To deal with the additional tasks, most of the teams have one team member responsible for communicating with the rest of the organization. In company B, this is the tech lead. A tech lead from company B describes his role as a link between the team and the rest of the organization. If anyone wants to talk to the team, they often approach him. According to several informants across the companies, there is one particular reason why this is often the team or the tech lead's responsibility; individuals may find interruptions disturbing, and also difficult to know what to prioritize. A developer from company B states that being interrupted when writing code makes the tasks much more time consuming, as his work requires deep concentration. Further, a developer from company C states that having someone who filters out unnecessary information, making sure only relevant information is spread to the rest of the team, is relieving.

On the other hand, when describing how the business department in company B communicates with the team, a tech lead remarks that they sometimes talk directly to the team member responsible for the certain task:

> *They talk directly to those having the task at hand. Sometimes this is fine, but ideally they should involve the whole team, so that everyone knows what is going on.*
>
> *Tech Lead, B*

Several team and tech leads state that they try their best to shield their team from externalities, filtering out what they consider as unnecessary for their team to know or take part in. A team lead from company C states that he has become increasingly strict when it comes to what he is willing to interrupt the team for. Another team lead states that the shielding responsibility is the single most important task he has.

Communicating with the Customer

Communication with an external customer mainly applies to teams in company C, since they are consulting teams. A developer explains how this responsibility is assigned to the team lead to avoid problems due to lack of technical knowledge:

> [My impression is that] *most of the communication would be the customer's responsibility if we did not assign it to the team lead in our team. That would have been problematic, as the customer does not have the same technical insight as a developer who is deep into the project.*
>
> *Developer, C*

Furthermore, a team lead from company C states that having only one person responsible for communicating with the customer makes the process much easier. This is supported by a developer in the same company, referring to an earlier experience of chaos when all team members had customer contact. Having only one team member responsible, the customer and the team know who to contact regarding for example questions or changes. The team lead can then spread the message to the rest of the team.

5.3 Intra-team Coordination

Planning and Communication

Several informants mention the question of planning. In company A, a product owner expresses frustration over the lack of plans in the team, and how working in an agile team makes it harder to plan in general. This is supported by a business representative from company B, who is responsible for planning and progression. He finds it comfortable to be in control and knowing who does what, but still tries to adapt to the new way of working:

> *For us who are responsible for progression, it is very comfortable knowing exactly who will take on the upcoming tasks in the next days. But I guess we just have to stand tall and hope it all settles as time passes.*
>
> *Business Representative, B*

Furthermore, a domain architect from company A expresses the importance of having an overall plan for the team's work, and that they are in the process of making one. Similarly, a tech lead from company B explains how new deliveries requires a lot of preparatory work and analysis. He highlights the need for balancing the planning process ahead of delivery versus making adjustments along the way:

> *The problem is that in this case* [a specific delivery] *you could have done more preparatory work, and figured things out upfront. And there is a balance between how detailed the planning process should be, and what adjustments should be made along the way.*
>
> *Tech Lead, B*

A business representative from company A explains that over the span of a week, there are multiple work meetings where either all or a selected few of the team members participate. The team also has a planning meeting once a week, where the same members meet every time. The business representative explains that as an agile team, the work and planning processes differ. He states that since they are a team of 10-12 members, they need a visual view of what they are doing at every instance. In the past they were extremely detailed, estimating everything including time usage. He says there is less focus on planning today.

Furthermore, the same business representative explains that they have started to use a digital communication tool in order to save time, reduce the number of meetings and facilitate easier clarifications. Earlier, a lot of time was wasted writing emails and arranging meetings. The same tools are apparent in company C, where a team lead explains that the team uses Slack and, to some degree, emails. Similarly, a developer from company B points to a web solution improves communicating specifications internally:

> *I would say that the internal communication is challenging, especially when it comes to communicating specifications. But now we have gathered all the specifications in a web page. That makes it a lot easier.*
>
> *Developer, B*

While this developer thinks communication is getting better, another developer from the same company claims that the communication can improve, especially when it comes to the design process and clarifications concerning their interrelated tasks. He claims that, as a developer, you do not always reach through with any objection you might have. He also states that this is a recurring issue, leaving a potential for improvement of the communication and coordination.

Work Methods and Ceremonies

All the teams practice some sort of agile methods. A tech lead from company B talks about the Kanban-like process they aim for in the team, and its benefits:

> *I do not know if we can really call it Kanban (...) but it is what we try to accomplish (...) it is what we try to do. When we make it work, it is beneficial, and the flow of tasks is faster. You can better follow the task from beginning to end.*
>
> *Tech lead, B*

A team lead from company C claims the method they are using is a hybrid between Scrum and Kanban. He elaborates on how the process unifies the team by setting a focus each week, somewhat similar to a sprint. He stresses how this is necessary, when working on different parts of a particular delivery consisting of interrelated sub-tasks.

The same team lead also explains that stand-up is used every day as an important practice for updating each other. However, he also emphasize the spontaneous dialogues emerging between the team members when clarifications are needed. Another team lead regards stand-up as a communication tool that he uses to update the team with information from the external stakeholders:

> *It* [the stand-up] *is mostly about communication from me to them, because I do most of the meetings with external stakeholders.*
>
> *Team lead, C*

This is also the case for another team in company C. In company B, stand-ups are also present. A business representative from company A does a weekly stand-up with the team, and uses it to jointly prioritize work.

Process Improvement

In company B, a tech lead claims that retrospectives generally lead to beneficial adjustments in the way the team works. A developer in the same company says the team conducts weekly retrospectives, where the members discuss both the positive and negative sides of their work.

In company A, a business representative explains that they just started using retrospectives. The team used to have a similar meeting once a month. A team lead from company C explains that they arrange retrospectives approximately every six months. Two other team leads from the same company state that they use retrospectives every now and then, where the focus lies on providing feedback to individuals and to the team.

A tech lead from company B agrees that direct feedback is useful, and that the team needs to focus on process improvement to function better. However, he states that it is not easy to find the time needed for process improvement. Being a developer in addition to being the tech lead has left little time for other assignments beside the technical tasks. This comes at the expense of the processes within the team. He further states that more time should be spent ensuring that the team is functioning well. A developer from company B reflects on how process improvement is not very structured:

> *Process improvement does not appear that structured to me. Often we just casually discuss what is working well and what is not.*
>
> *Developer, B*

This is similar to another developer's comment on the topic, stating that they do not discuss processes often enough. He agrees that this is mostly a matter of limited time, and that he would prefer to discuss improvements more frequently. Additionally, a tech lead from the same company explains that there is no time for process improvement when the focus lies on making deliveries. He states that it can be difficult to do something about the process when the road to delivery is already set.

5.4 Decision-making

Prioritization and Distribution of Tasks

For company A, a business representative explains that several members of the team participate in a weekly meeting concerning prioritization and distribution of the team's tasks. This is followed by a meeting, gathering the whole team to present the decided prioritization:

> *We set aside half an hour with the team where we present the priorities of the week, in order to make everyone aware of what tasks they should prioritize.*
>
> *Business Representative, A*

In company B, the product owner and a representative from the business department prioritize the team's tasks. In company C, a team lead states that the team members collectively prioritize tasks in his team. He thereafter coordinates the distribution of tasks, by letting the team members choose the tasks they want from a list:

> *When a new assignment comes in, I split it and distribute it to the team. Or, actually I say "These are the tasks, who wants to do what?" or "Here is a list of tasks, pick something, please".*
>
> *Team Lead, C*

Confirming this, other informants from company C express that the team members mostly have the opportunity to choose which tasks they want to solve themselves. A developer states that during hectic work periods, the team members are often assigned tasks by the team lead. In other cases, the team members choose what they want to work with from a list.

A line manager in company B explains that in some teams, the tech lead has an open dialogue with the team members in order to clarify who wants to do what based on their competence. In another team, the informant's impression is that the representatives from the business department assign the tasks to the team members. The line manager further expresses that the team ideally should distribute their own tasks without involvement from the business department. However, a statement from a tech lead from company B shows how business representatives are closely involved in this today.

A business representative from company B expresses that he finds it challenging to identify what should be prioritized in the task distribution; the individual learning of the team members, or the team's progress. He states that the goal is to let the developers decide what tasks they want to solve themselves, but that he sometimes has to micromanage at an individual level in order to reach the team goals.

Decision-making Authority

A business representative from company B expresses that he tries to give the team members more decision-making authority, as he believes that people do not feel ownership of a task if they get told what and how to do things. Still, an informant expresses how business representatives are highly involved in the team's decision-making:

> *The business representative plays a decisive role. He has the final word when deciding the outcome. So, yeah. He plays a decisive role in how decisions are made.*
>
> *Developer, B*

However, there are also examples where the team members are more involved in the decision-making processes. A tech lead from company B expresses that all team members can affect the decisions in the team if they want to. Another tech lead from company B exemplifies this, stating that the team members often have the opportunity to share their opinions on an issue with the product owner before he makes a final decision:

> *The developers have the opportunity to say “this will not function well in practice”, or “this should be changed”, and it will be prioritized in the same way as other suggestions. But at the end of the day the product owners decide the prioritization of these suggestions.*
>
> *Tech Lead, B*

A team lead from company C expresses that the team usually reaches consensus together, and that it has never been necessary to make the final call, even though he has the authority to do so. As the teams in company C are consulting teams, they are closely connected to external customers. As explained by a line manager, it is the customer who requests the product and pays for it, and the team has to do what the customer wants. Although the decisions in most cases are made collectively by the team, the customer possesses the final decision-making authority. A developer describes this as being autonomous given the customer’s priorities:

> *I will say that we are pretty autonomous, most of the time. Under the priorities from the customer, though.*
>
> *Developer, C*

A team lead in the same company agrees, explaining how the team’s autonomy is dependent on how detailed the control from the customer is, which he argues in turn depends on the level of trust between the team and the customer. When the trust is being challenged for some reason, he tells that the customer might want to manage the team in a detailed way. This becomes another source of frustration, as the customer often has limited technical insight and therefore a poor basis for finding the best solutions. Developers express that it can be demotivating when the customer decides on something that the team members see as a less appropriate suggestion. A developer from company C elaborates on this challenge, highlighting the frustration when the customer wants a solution that is technically not a good idea:

> *Another challenge is when the customer wants something, and you do not understand why or how they want it. That is a frustrating challenge; when the customer thinks it is a good idea, but technically it is a very bad idea.*
>
> *Developer, C*

Training

Several informants from the three case companies state that they did not get any training for their new roles or in making shared decisions. A developer from company B got assigned the tech lead responsibility, but did not get any training. He explains that he would like to know more about how to perform his role:

> *Sometimes I think that maybe I should know more about what I am supposed to do. Maybe I am not doing enough for my team.*
>
> *Tech Lead, B*

A team lead from company C explains how he started working as a team lead without being explained how to perform his role. The empirical data shows how some leadership roles include spending a high percentage of the time administering the team and participating in meetings. Informants express these tasks as new to them, and some admits that these are tasks they do not want to have. With a lack of training, some interviewees explain this as a burden, while others see it as a new and interesting challenge.

5.5 Human Factors

Predefined Roles

A line manager from company B expresses uncertainty about either defining and assigning roles to individuals, or defining responsibilities on a team-level leaving how to handle it to the team:

> *It is difficult to know whether you should have very clear role definitions, or if you should define the responsibilities of the team and let them decide how to solve it.*
>
> *Line Manager, B*

Several informants state that the leadership roles within the teams are assigned to team members by higher level managers when the teams are initiated. Furthermore, an informant from company B expresses doubt about the actual need for some of the existing leadership roles. He states that not all of the roles are well adapted to work in an autonomous team. This is supported by a line manager from company B, explaining how the descriptions for some of the leadership roles have not been adjusted to the new work design. Similarly, a product owner from company A explains how the organization is not structured optimally for autonomous teams. The informant states that many middle managers have been assigned different roles within teams, which he considers less optimal.

Further, the need for leadership roles within the teams are discussed by several informants. A developer from company B explains that there is a need for what he calls "a captain" holding the overall responsibility, and someone with a final decision-making authority. However, he is not sure whether it is really necessary to organize it this way, but concludes that this is how they do it. A product owner from company A reflects on the situation in his team, and states that a strong internal leader is not needed:

> *There are two kind of leadership roles in my team. I think an additional stronger and more traditional leadership role would have been very unnatural.*
>
> *Product Owner, A*

Undefined Responsibilities

In many of the teams, only a few of the roles are clearly defined with assigned responsibilities or role descriptions. In other cases the roles are seen as being just titles. Informants from company B express bewilderment regarding leadership roles and who is responsible for what. There are also examples of frustration regarding who is responsible for the deliveries in a team, when it is supposed to have flat structure. Those who carry the responsibilities also state that they are frustrated. An example is a product owner from company A who expresses a lack of control and power to influence his team:

> *I have no control over what they do all the time. It is not like I need that, but I have no power to get things I see as important through in the team.*
>
> *Product Owner, A*

Some team members with leadership responsibilities from company B claim to have seen a brief description of their new roles once, while others have never seen it at all. A tech lead from the same company explains how the tech lead role was formed during the reorganization of the company. The informant explains how he got a description of what his new role concerned, but that he does not remember all of the defined responsibilities. The complexity of the tech lead role is supported by a project manager from company C, stating that it takes a lot of time to understand what the tech lead role is really about.

Competence

A product owner from company A claims that the team needs the right competence. However, he is not sure if his team is designed and structured well according to what the team is supposed to develop, or if necessary fields of knowledge are covered by the team members.

In company B, a developer highlights the high level of competence in his team, claiming that the team members are highly skilled. A team lead from company A makes the same claim, and elaborates on how different members having different skills comes into play when distributing the tasks. He explains that tasks are normally distributed with regards to competence, and that it is often obvious who should be assigned which tasks.

Furthermore, a developer from company A expresses doubt concerned with how well the different fields of knowledge come together in cross-functional teams. Some roles with leadership responsibilities are responsible for aspects they do not have knowledge about. The developer therefore stresses the need for technical understanding among leaders. When business managers are set as responsible for technical deliveries, some of the developers feel that the responsible manager has a lack of understanding of technical challenges and how to approach them.

6 Discussion

This chapter discusses our findings, as illustrated in Figure 6.1. We analyze them in light of relevant theory presented in the theoretical background, and identify challenges for the teams and how they relate to autonomy.

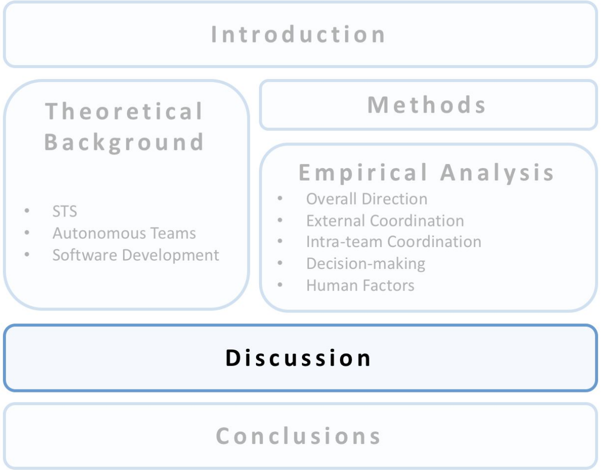


Figure 6.1: Structure of the discussion.

To recall, the five themes from the TA is shown in Figure 6.2. The discussion of the findings from each theme is presented in the same order as presented in Chapter 5.

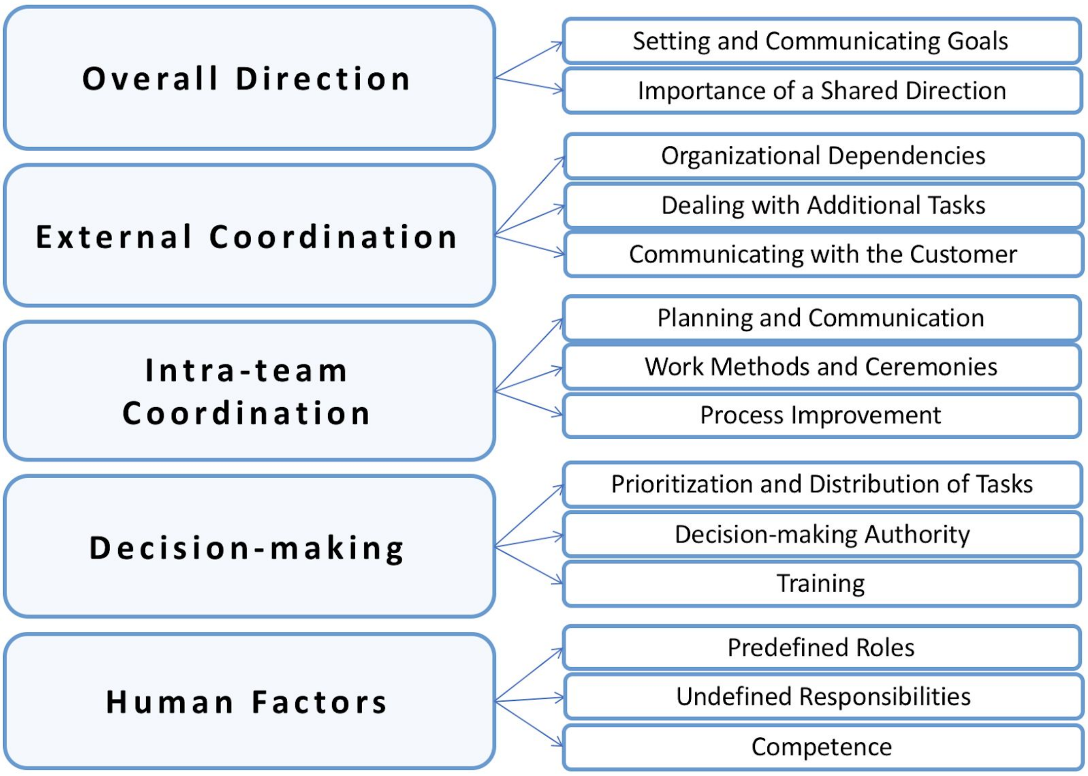

Overall Direction
Setting and Communicating Goals
Importance of a Shared Direction
External Coordination
Organizational Dependencies
Dealing with Additional Tasks
Communicating with the Customer
Intra-team Coordination
Planning and Communication
Work Methods and Ceremonies
Process Improvement
Decision-making
Prioritization and Distribution of Tasks
Decision-making Authority
Training
Human Factors
Predefined Roles
Undefined Responsibilities
Competence

Figure 6.2: Overview of the themes in the empirical analysis.

6.1 Shared Goals and Direction - to Where?

Section 5.1 reveals how goals are set by higher level managers, and are not always successfully communicated to the team. In this section, we will discuss the topic of committing to shared goals and direction.

Commitment to Goals

The empirical analysis shows that team members are often or always excluded from goal setting processes. Rather, the goals are set by higher level managers and are given to the teams. This contrasts Manz & Sims' (1987) view, who state that external leaders should allow the teams to set their own goals in order to facilitate autonomy. Participation in goal setting is also associated with an increase in motivation (Hackman & Oldham, 1980), and increased meaningfulness for those who are trying to achieve them (Hackman, 1987), since the goals are less trivial. Despite the benefits of letting the teams participate, this is evidently not the approach in the teams we have studied.

Furthermore, a business representative states that the goals are communicated well, and his impression is that the team members have an understanding of where the team is headed. However, the empirical analysis shows that this is not always the case. Even though both business representatives and developers regard goals and direction as important, there is not necessarily a shared understanding of them. This is illustrated by informants having different views on goals. One view is that goals are closely related to deliveries, e.g. finishing something in time. A second view is that goals are set so that teams have something to strive for, not necessarily something to achieve in a certain time. A third view is that higher level goals are not communicated in such a way that they reach the team level. This is apparent from an informant not really knowing what the higher level goals are.

In other words, there is an incongruence; while the impression among leaders is that goals are communicated and understood by the team, statements from team members indicate that this is not the case. There seems to be a lack of shared understanding of the goals set by the management and what direction to take. According to Moe et al. (2009), a challenge that follows from this is individual commitment; team members will pursue their own goals if they have no reason to commit to the shared ones. Hence, if higher level managers do not let team members partake in setting goals, the team members might create and pursue their own goals instead.

Therefore, not letting the team partake in goal setting will likely impact the team's autonomy in two ways. First, if not participating in the goal setting causes the team members to set their own goals, the individual autonomy will increase, as individuals are working independently toward their own objectives (Langfred, 2000). Second, according to Hoegl & Parboteeah (2006), the external autonomy will decrease, as the team does not have the authority to decide its own goals. The goals are set by higher level managers, deciding what is important for the team.

To summarize shortly, commitment to goals set by higher level management is a challenge for autonomous development teams. This can increase the individual autonomy, since team members can potentially set their own goals. It can also reduce the external autonomy, since the team does not participate in setting their own goals.

6.2 Shielding the Team

The teams in our study are all part of larger organizations. Section 5.2 reveals how the teams need to coordinate and communicate with their external surroundings. This section explores this topic further.

External Dependencies

The empirical analysis shows that the teams communicate and coordinate interdependently with other teams and departments within their companies, or with a customer and other external teams that are involved in the same development processes. This seems necessary for two reasons. The first is that the specifications of the product, such as new requirements, are subject to change over time, and therefore need to be communicated to the team. Secondly, resources often have to be synchronized between multiple development teams, as the final product can depend on components from many of them.

However, Pikkarainen et al. (2008) state that agile practices do not provide the communication mechanisms in situations where many teams are involved in the same development process. In practice, a common solution seems to be that higher level managers assign the responsibility of the external coordination to a leadership role within the team. From this way of dealing with the external environment, we draw a parallel to what Boehm & Turner (2005) refer to as a project manager; a role operating as a barrier between the organization and the team. Despite the fact that the case companies have different titles for the role responsible for the external coordination, all of them seem to have one aspect in common; they assign the responsibility to *one* designated team member with leadership responsibilities*.* Depending on the case company, this responsibility is assigned to either the team lead, tech lead or product owner.

According to informants, the person responsible for the external coordination is tasked with shielding the team. This involves protecting the team from unnecessary interruptions, and deciding which pieces of information that are important enough to put forward to the team. Empirically, those who have such a role consider themselves as links between the team and other departments of the organization. In cases where the team relates to an external customer, they take care of the communication and information flow between the customer and the team. The general need for coordination is addressed by Nerur et al. (2010), stating that software development teams need to interact with an ever more diverse set of stakeholders, who have different expectations and needs of the team. In other words, shielding the team is a complex task.

Furthermore, the empirical analysis also shows that teams frequently receive additional tasks from their external surroundings. These tasks are often outside the scope of what the team is assigned to do, such as tasks concerning errors in previously delivered products. According to Susman (1976) in Manz & Sims (1987), a team is more able to reallocate resources in changing work conditions compared to individuals in a rigid structure. Also, agile teams are more flexible and respond more rapidly to complex and ever-changing problems (Cockburn & Highsmith, 2001; Dybå, 2000). However, the empirical analysis reveals how the additional tasks delay the teams in their own work, since they are forced away from what they initially were doing. In that sense, the adaptability and flexibility may itself impede the team's progress, as considerable capacity is used to solve unrelated tasks.

Hence, having someone shielding the team from external surroundings seems to be important. The empirical analysis reveals that developers find the shielding role relieving, since getting interrupted while focusing on the work make tasks more time-consuming than they need to be. However, the empirical analysis shows that even though the teams have someone to shield them, the information and distribution of additional tasks coming from the surroundings do not always go through this contact point. Sometimes, representatives from various organizational departments and customers approach developers in the teams directly. This is similar to what Moe et al. (2008) explain as *stealing* resources; external stakeholders, such as customers, approach and occupy developers with unrelated tasks. In other words, external stakeholders approach team members directly, despite members expressing that it disrupts the work they are originally assigned to do.

Based on the empirical analysis, we see two possible reasons for why the contact point is bypassed. The first reason is that the contact point might be a bottleneck of information. One of the holders of the shielding role states that he spends most of his time in meetings and on administrative tasks. He might not be available, or simply be overloaded with information. By having only a single point of external information in the team, the contact point may be exposed to what Schick et al. (1990) describe as information overload; too much information to handle. This means that relevant messages might disappear in the overflow of information. As a result, external stakeholders might see it as more reliant to approach the team members directly. The second reason is delivery-focus. As the empirical analysis shows, the teams have tight deadlines and many intervals of work. They are therefore time-sensitive, and external stakeholders who have their own deadlines, might not be willing to wait for a response when they can just approach the team members directly to get what they want.

Thus, the discussion reveals contradictory interests; the external stakeholders want to make use of the team's resources and make ongoing clarifications, while team members prefer being shielded from external noise as it interrupts their work. If the shielding role is bypassed, the team's control over their work is limited by the involvement from the external surroundings. According to Hoegl & Parboteeah (2006), the external autonomy is therefore reduced. Also, if individuals are assigned tasks directly by the external environment, their freedom and control in carrying out their own tasks is impeded. Therefore, the individual autonomy is reduced (Langfred, 2000).

To summarize, this section identifies the challenge of external dependencies, which requires that the team communicates and coordinates with external surroundings. This challenge is approached by assigning the responsibility of external coordination to a team member with leadership responsibilities. However, external stakeholders approaching team members directly with additional tasks can lead to a reduction in external autonomy, since the team's control over their own work is limited. This can also reduce the individual autonomy, since the team members are no longer responsible for, or free to carry out, their work.

6.3 Is Coordination the New Planning?

When a number of people is working together on interrelated tasks, the need for coordination increases (Langfred, 2005). Section 5.3 presented the level of planning, need for shared work processes and communication, as well as improvement of these processes. This section explores the topics further.

Coordination Mechanisms

The teams in this thesis develop complex software solutions. The empirical analysis shows that such products are a result of interrelated sub-tasks, and that the team must be aligned in order to form a complete technical solution. Langfred (2000) stress how working in a team with interrelated tasks implies a need for coordination and knowing what the others in the team are doing. However, informants seem to have contrasting opinions on whether coordination of work is a planning requirement, or rather a communication and transparency issue. More specifically, the empirical analysis reveals mixed perceptions on what the informants consider should be planned and specified. There seems to be two distinct views on this. The first view is presented by non-leaders in the team, mainly developers, expressing that an overview of who is doing what is a sufficient way of planning. The second aspect is presented by certain informants with leadership responsibilities, expressing frustration over the lack of planning, and requesting more control over tasks.

The first view indicates how developers seem to have less focus on plans, and rather stress the need for coordination of interrelated tasks. The empirical analysis reveals that several teams use Slack as a communication tool, and that some use boards or digital planning tools to illustrate and keep track of what they are doing and where they are in the process. They also conduct agile ceremonies for coordination such as stand-ups, which are meant to be short, daily meetings to report progress, impediments and track progress (Stray et al., 2016). Informants refer to these meetings as an important part of the communication and coordination, and as an arena for task-specific clarifications in a shifting environment.

In contrast, the view presented by a number of managers and leaders, emphasizes planning and control of the teams' tasks. Some leaders seem to hold onto the mindsets emphasized by former approaches to software development, such as the waterfall model. In other words, they still stress extensive and detailed planning of activities within the teams, and focus less on coordinating tasks. According to Balkema & Molleman (1999), leaders can struggle with the notion of self-management, and keep planning for and monitoring the team in the same way as before. If leaders believe that the team is unable to manage itself, they might feel a need to intervene in the team's processes.

As a consequence, the empirical analysis shows how the stand-ups frequently become more like meetings where team members update leaders on the team's progress, than arenas for coordinating interdependent tasks.

According to Stray et al. (2016), the focus on updating leadership roles inside and outside of the team on how they are doing, is seen to impede the intended focus on keeping the other team members up-to-date.

In other words, there is an incongruence between some leaders' need for plans and control, and the developers' need for coordination to solve their interrelated tasks. The incongruence might come at the expense of the autonomy of the team. If the push for planning originates from external managers exercising close supervision of the team, the external autonomy of the team decreases (Hoegl & Parboteeah, 2006). This happens when external stakeholders exercise control over the team, requiring more plans than the team considers necessary. Further, by impeding in the developers' focus on coordination, the decision-making can be distributed to individuals. Hence, the individual autonomy increases, as the team members find their own ways of solving the tasks (Langfred, 2000). In turn, this may reduce the incentives to engage in shared decision-making, decreasing the internal autonomy of the team (Moe et al., 2008).

Process Improvement

The empirical analysis shows how several developers highlight the importance of process improvement in the form of conducting retrospectives. According to Gonçalves & Linders (2013), these are intended to be an arena for reflecting on past work processes and discussing how the team can improve. However, despite the developers' emphasis on improving team processes, informants experience the retrospectives as being both few and far between, and conducted in an unstructured way. Often, as explained by several informants, this is caused by the difficulty of finding available time, given the teams' tight deadlines and continuous small deliveries using agile methods. This is in line with Dybå (2000) and Nerur & Balijepally's (2007) description of agile methodologies.

The empirical analysis also reveals differences in learning outcomes depending on how the retrospectives are carried out, and how the results are implemented. While some informants describe it as a valuable practice that their teams improve a lot from, others admit that the discussions often end up with suggestions no one brings into action. This is in accordance with Gonçalves & Linders (2013), who indicate that such ceremonies might become a waste of time if they are not conducted properly.

The empirical analysis also shows that higher level managers do not seem to notably encourage the teams to increase the frequency of retrospectives. This is in contrast to Gonçalves & Linders' (2013) view that frequent retrospectives are proven to generate business value in the long run. Also, Moe et al. (2009) identified *failure to learn* as impeding for self-managing teams. By not conducting retrospectives, the teams will struggle with improving their processes. Hence, the lack of emphasis on retrospectives from higher level management can affect the teams' possibility of becoming autonomous.

As shown in the empirical analysis, the teams are working under high delivery pressure from higher level managers, moving from one delivery to the next. As a consequence, when higher level managers do not provide the team the opportunity and authority to improve its processes, the team's external autonomy is reduced (Hoegl & Parboteeah, 2006). Furthermore, the informants consider retrospectives as an important arena for providing feedback and improve as a team. However, when these ceremonies are given a low priority and not conducted as intended, the teams do not get the chance to improve their processes. Hence, they do not get the opportunity to improve their shared decision-making processes, which might reduce the internal autonomy (Moe et al., 2008).

To summarize shortly, this section identifies two challenges. The first is coordination mechanisms, and concerns the leaders' need for planning impeding with the team’s need for coordination of interrelated tasks. This is closely related to the autonomy of the team; it increases the individual autonomy of the team, and reduces the internal and external autonomy. The second challenge is the lack of time for process improvement. When the higher level managers do not give teams the time for process improvement, the external and internal autonomy is reduced.

6.4 Deciding How to Decide

By definition, members of self-managing teams are responsible for planning, executing and monitoring their own processes and tasks (Goodman et al., 1988). In this section, two topics will be discussed. The first regards how individuals with leadership roles influence the decision-making processes in the teams. The next concerns how the team is affected when a customer has the final say in the teams’ decision-making.

Unequally Distributed Decision-making Authority

The first topic concerns how the decision-making authority seems to be unequally distributed among the members in some of the teams; the authority is found to be exercised by individuals with leadership responsibilities. Decisions on important aspects such as prioritization and distribution of tasks are often made without much presence of the other team members. Such a decision-making process evidently goes against the principle of shared leadership (Carson et al., 2007; Moe et al., 2009), as the team members do not share the decision-making authority.

Why some leaders, deliberately or not, play decisive roles in the teams’ decision-making processes, can be understood by the context of the teams. The empirical analysis shows how the teams work under a high delivery pressure from higher level managers or customers. Some leaders react to this by centralizing the decision-making structure with the intention of ensuring progress in the teams. Hence, assuming that shared leadership is sought after in agile teams (Moe et al., 2008), the leaders' strong positions in decision-making can be an inhibiting factor. This is because most team members are not included in the decision-making processes.

Another possible reason for leaders exercising decision-making authority in the teams, can be a lack of understanding or knowledge on how to participate in shared decision-making. This is in line with Moe et al. (2009), claiming that team members do not automatically know how to work together as a team. The lack of dedication to shared decision-making can therefore be a consequence of leaders not knowing how to approach their new roles. Also, according to Langfred (2007), and Manz & Sims (1987) in Balkema & Molleman (1999), teams will not achieve self-management without training and support of their members. Despite this, the empirical analysis shows that individuals with leadership roles have not received any extensive guiding on how their roles should be adjusted to working in autonomous teams. As a consequence of team members not receiving training, there is a possibility of decisions being made in the same way as before autonomous teams were introduced, as found by Stray et al. (2011) and Moe et al. (2009).

This discussion reveals a paradox; the benefits associated with autonomous teams are desired, but managers still fall short in providing necessary training and support to the teams and their members. A possible reason is the emerging confusion regarding the term and concept of autonomy, presented in the introduction of Chapter 5. The empirical analysis shows that informants at different levels in the organization seem to have divergent understandings of autonomous teams. Thus, without agreeing on what they are aiming for, it is probably difficult to achieve it.

This limited understanding of autonomy might be the reason why individuals keep making decisions for the whole team. Empirically, a centralized decision-making structure has shown to be disadvantageous for the technical solutions being developed. A developer explains how a technical solution ended up being suboptimal, because important considerations and suggestions from the developers were not taken into account by leaders making the decisions. In terms of autonomy, centralized decision-making is contradictory to internal autonomy (Moe et al., 2008), as the decision-making authority is not shared among the team members. Subsequently, the internal autonomy is low.

Customer Authority

The second topic concerns teams working for external customers, namely consulting teams. The empirical analysis shows that the team members in these teams participate in decision-making to a high degree. Several informants highlight that prioritization and distribution of tasks is considered a shared responsibility among team members. However, regardless of how the decisions are made, the customer makes the final call of how the delivery should be formed, even if the team disagrees with the customer's choice.

This is found to have several implications for the teams' autonomy. When decisions regarding prioritization and tasks are a result of shared decision-making, the internal autonomy can be increased (Moe et al., 2008). However, when the final decision-making authority is outside of the team's control, the external autonomy is decreased (Hoegl & Parboteeah, 2006). According to Tata & Prasad (2004), if a team is given limited authority from its customer, the team will only experience symbolic self-management; what they jointly decide may not really matter for the final outcome.

Furthermore, an informant stresses trust in the relationship with a customer. He claims that the level of trust is closely connected to the degree of control the customer exercises over the team. This is in line with Hoegl & Parboteeah (2006), who state that the influence from external actors on the team's decisions may be a signal of their level of trust to the team. Langfred (2000) also supports this, claiming that trusted teams are likely to be less controlled and monitored. In other words, the team's external autonomy, as defined by Hoegl & Parboteeah (2006), is proportional to the level of trust from the customer.

To summarize shortly, this section identifies two challenges. The first concerns unequally distributed decision-making authority, which shows how delivery pressure and a lack of training can contribute to decision-making processes being centralized to leadership roles. Hence, shared leadership and the internal autonomy of the team is reduced. The second challenge is customer authority, which entails that when a customer is present, the final call in decisions lies with them. This decreases the external autonomy of the team, and can lead to a team only experiencing symbolic self-management.

6.5 Roles and Responsibilities

There are two interesting topics emerging from Section 5.5 that will be explored in this section. The first is the bewilderment regarding roles and their titles. The second is the specific skills associated with each member in the team.

Empty Role Titles

The empirical analysis shows that higher level managers define leadership roles and assign them to certain team members before the team comes together. According to Hackman (1987) and Stewart & Manz (1995), defining roles should be left to the team. They see role-defining as an important development process that can lead to team members feeling attached to the self-assigned responsibilities, and role definitions that correspond to the team's needs.

In light of this, several informants reflect on the leadership roles in their teams. They point out that the pre-assigned roles are rarely more than just a title; exhaustive role descriptions and associated tasks are seldomly included with the titles provided. We also see several examples of role titles not adapted to the new way of working in teams, such as some of the business representative roles. Moe et al. (2009) found that Scrum Masters ended up making decisions just like they did as traditional managers, especially when the team faced problems. In other words, if a role or title is similar to one present before adapting to autonomous teams, it can be easy to hold on to former habits.

Similarly, for a team member holding a leadership title without defined responsibilities, it might be easy to take on the decision-making authority associated with being a leader. This type of authority resembles what Bolman & Deal (2009) define as power of position; authority following from a given title or role. If the source of power is recognized by the rest of the team, the authority of the individual becomes significant (Bolman & Deal, 2009). As a result, the decision-making authority is associated with individual power as in a centralized decision-making structure. Thus, the internal autonomy of the team is reduced (Moe et al., 2008). Also, refraining from letting the team define their own roles from team-level responsibilities can be seen as a reduction in the external autonomy (Hoegl & Parboteeah, 2006).

Specialists

All teams in this thesis are defined as problem-solving teams; they combine a variety of skills and knowledge to produce something unique (Ratliff et al., 1999). As pointed out in Chapter 4, many of the teams are cross-functional, and they cover a variety of business and software functions in order to develop a product in the intersection of the two disciplines. This is in line with Morgan (2006) and Parker & Wall (2006), who state that autonomous teams should have all necessary skills and knowledge to fulfill its goals within the team.

Furthermore, Moe et al. (2009) claim that an autonomous team needs generalists to ensure redundancy; members that are multi-skilled and able to perform the jobs of others if needed. However, the empirical analysis indicates that the team members are not multi-skilled. Informants view their own roles and functions as very different because they are separate professions; they see themselves as specialists, not generalists. This is supported by Uhl-Bien & Graen (1992), claiming that members of cross-functional teams are not able to cover each others' work, as they are not of the same discipline. In other words, business developers can hardly cover the job of software developers, since they are not interchangeable fields of knowledge.

Hence, while the teams are cross-functional, the team members are not necessarily multi-skilled. A consequence of this, is a lack of understanding of each others' field of knowledge. For instance, some of the developers state that if a manager with a business background is responsible for a technical delivery, they notice a lack of understanding of the assignment and how it should be approached. This might be a result of his business background and limited technical insight. Multi-skilled or not, Moe et al. (2009) claim that the team members need to understand what the others in the team are doing if they are to align their individual tasks and decisions. According to Ratliff et al. (1999), it is the combination of the team members' skills that results in the desired product. Therefore, even though the team members have different backgrounds and do different work, they still need to combine their knowledge and make shared decisions.

To facilitate joint problem-solving processes, Hoegl & Parboteeah (2006) emphasize openly sharing technical and coordinative information. If a team does not succeed in combining the team members' knowledge, it is hard to make the intended and best possible product. With specialized team members, such sharing of knowledge can be seen as vital to adopt shared decision-making, and thereby increase the internal autonomy of the team (Moe et al., 2008). Subsequently, not sharing knowledge can result in a decrease in the internal autonomy. Another possible outcome from the high degree of specialization, is individuals being the only ones understanding their tasks. This leaves them with a high individual autonomy (Moe et al., 2008), as they have freedom in carrying out their work.

To summarize shortly, this section identifies two challenges. The first is empty role titles predefined by higher level management. This reduces the external autonomy, as the the team does not have the authority to define their own roles. Furthermore, the internal autonomy is decreased by centralized decision-making. The second challenge concerns teams of specialists. By not sharing their specific knowledge, it becomes difficult to make shared decisions. Hence, the internal autonomy is reduced. In addition, for a team of specialists possessing knowledge of separate fields, the members are free to carry out the work as they see fit. This results in an increased individual autonomy.

6.6 Summarizing the Discussion

In order to answer our RQ, we have identified eight challenges and described their relation to autonomy. The first challenge concerns a lack of *commitment to goals* set by higher level management. This can increase the individual autonomy, since team members potentially set their own goals. It can also reduce the external autonomy, since the team does not participate in setting their own goals. The second challenge is *external dependencies*, which entails that the team has to deal with its environment. If the team's and its individuals' control over their own work is limited because of their external dependencies, the external and individual autonomy is reduced.

Coordination mechanisms is the third challenge, as the leaders' need for planning impedes the team's need for coordination of interrelated tasks. It relates to autonomy on the individual, internal and external level. The fourth challenge is lack of time for *process improvement*. If the higher level managers do not give teams time to improve their processes, the external and internal autonomy is reduced.

The fifth challenge is *unequally distributed decision-making authority* within the team. This can reduce the internal autonomy, caused by high delivery pressure and lack of training in decision-making. *Customer authority* is the sixth challenge. When a team relates to a customer, the final call when making decisions lies with them. Hence, the external autonomy is reduced.

Empty role titles predefined by higher level management is the seventh challenge. This can lead to a reduction in external autonomy, since the team does not have the authority to define their own roles. Also, the internal autonomy is decreased if empty role titles causes centralized decision-making. The eighth challenge concerns teams consisting of *specialists*. By not sharing their specific knowledge, it becomes difficult to make shared decisions, resulting in a reduced internal autonomy. Since the members are free to carry out the work as they see fit, the individual autonomy is increased.

7 Conclusions

This final chapter concludes our research, as illustrated in Figure 7.1. After the conclusion, limitations of the study are presented, followed by implications and an agenda for future research.

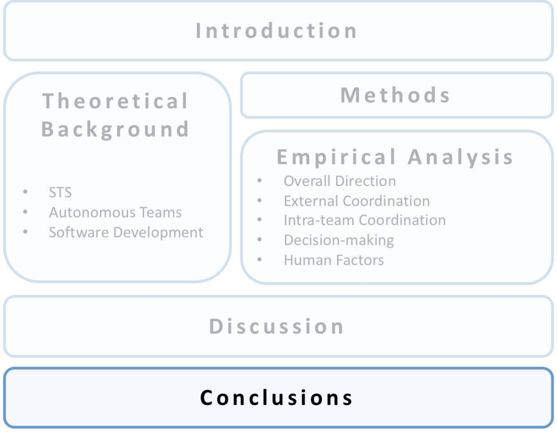


Figure 7.1: Structure of the conclusions.

7.1 Conclusion

This thesis identifies challenges for autonomous developments teams, and how they relate to autonomy. Adapting to autonomous teams is known to be difficult, and as a different way of working there are several issues to deal with. Using an inductive approach we identify the following challenges for autonomous development teams: commitment to goals, external dependencies, coordination mechanisms, process improvement, unequally distributed decision-making authority, customer authority, empty role titles and specialists. Some of the identified challenges are related to organizational conditions, while others are connected to the team's internal processes or specific roles within the teams.

Autonomy appears to be a complex, stratified term. In this thesis, the concept is regarded as having three distinct layers: autonomy on the individual level, the team level, and the level given by the team's external environment. We find that all identified challenges relate to one or more of these levels of autonomy. Depending on how the challenges are approached, the related autonomy levels increase or decrease. The takeaways from this study is partly the identification of challenges, but mostly the exploration of their relation to autonomy. There seems to be a limited understanding of the terms and processes associated with autonomous teams, and autonomy on different levels. Our thesis provides knowledge on how managers and team members can make conscious choices regarding autonomous teams. It contributes to the understanding of how dealing with the challenges impacts autonomy in different ways.

7.2 Limitations

This thesis has several limitations. First, a potential limit to our research is the fact that we studied the teams at a single point in time or over a very short timespan. Due to the time constraints of the master's thesis, a longitudinal study was not possible. Additionally, some of the teams were undergoing structural changes at the time of our data collection, and the effect on our research and conclusions remains unknown. The time constraints also limited the possibility of getting a deeper insight into all the teams in the three case companies. On the other hand, the teams under study were at different stages in the adapting to autonomy, which may counteract this limitation.

Furthermore, the majority of our informants had leadership responsibilities. Ideally, our sample should have included just as many informants without leadership responsibilities, in order to reduce the likelihood of a one-sided bias. Also, because of limited access, our sample did not include any external customers. Hence, when customers are discussed in this thesis, we have based our reasoning on how informants from the case companies present them.

There are also possible limitations to our analysis approach. The data was analyzed thematically, and several actions, such as recoding and exchanging samples, were taken to ensure reliability in the process. However, according to Bryman (2016), there is a chance of losing the social context of what is said in the coding process. Also, the narrative flow of what is said is lost due to fragmentation of data (Bryman, 2016).

In addition, we conducted an inductive multiple case study in order to answer our RQ. There might be variables that affect the challenges and autonomy that we are unaware of. Also, this inductive study does not provide an exhaustive list of challenges for autonomous development teams. Furthermore, the authors may be prone to confirmation bias, interpreting empirical findings in a way that confirms existing opinions (Nickerson, 1998).

The last limitation is related to generalization, and is a consequence of our sampling approach. All case companies (and therein teams) included in our sample, were already participating in the A-team project and chosen through purposive sampling. In light of this, the teams are to some extent interconnected, and operate under many similar organizational conditions. Our findings may therefore not be applicable to teams outside the context of our sample (Bryman, 2016). Yet, we consider our findings to be of relevance for other autonomous teams (more or less) similar to the teams in our sample.

7.3 Implications and Future Research

First, we want to underline the complexity of the term autonomy. Autonomy appears to be a stratified term, operating on different levels. This implies a need for further knowledge and insight into the different aspects of the concept, as well as awareness and an understanding of what to aim for with autonomous teams. We encourage more studies exploring the different aspects of autonomy.

We also see the need for an organizational commitment to autonomous teams, not just from the team members. Actors in the team's environment are closely related to the authority given to the team. When it comes to planning and control, we see signs of managers holding on to mindsets belonging to how software was developed using former methods. We argue that adapting to autonomous teamwork implicates a shift in the way of thinking, not only the way of working. In particular, we encourage higher level managers to let the teams participate in decision-making and set their own goals, as this appears to be vital for autonomy.

Another topic we suggest for further research, is roles and the allocation of responsibilities. There are different practices and solutions available, however, the literature on this is not extensive. We put forward questions such as how responsibilities should be specified and divided among members, whether or not distinct and defined roles is the best practice in the teams, and how specialized team members affect team performance and autonomy. Furthermore, the organizational structure should be explored more in-depth. We noted from our data that attitudes, strategies and structure of a company had significant effects on the performance of the teams.

In general, we suggest more qualitative case studies on autonomous development teams. More specifically, we emphasize the need for studies of multiple teams in their natural environments. There is a growing trend of autonomous teams, but the lack of empirical research seems to limit the understanding of the associated concepts. We hope for studies on a larger scale, with broader samples and a high potential for generalizability. Additionally, we consider research with a longer timeframe and closer interactions with the teams will give valuable insight. This study is only a snapshot, and it is likely to believe that the teams’ characteristics change over time. It could also be of interest to utilize different methods for data gathering, such as ethnography or shadowing. This will give a more realistic and nuanced picture of the teams compared to only conducting interviews.

Altogether, we regard our inductive study as new insight into how challenges relate to the autonomy of teams. Managers and team members can hopefully use this research as a foundation for knowledge creation, for making conscious choices, and for evaluating and improving existing practices. We hope this study can motivate and stimulate further research in the field, as an important implication of this study is the rise of new and unanswered questions.

8 Bibliography

Balkema, A., & Molleman, E. (1999). Barriers to the development of self-organizing teams. *Journal of Managerial Psychology*, 14(2), 134-150.

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

Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer,* 21(5), 61-72.

Boehm, B. W., & Ross, R. (1989). Theory-W software project management principles and examples. *IEEE Transactions on Software Engineering,* 15(7), 902-916.

Boehm, B. W., & Turner, R. (2003). *Balancing agility and discipline: A guide for the perplexed.* Boston: Addison-Wesley.

Boehm, B. W., & Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *IEEE Software*, 22(5), 30-39.

Bolman, L. G., & Deal, T. E. (2009). *Nytt perspektiv på organisasjon og ledelse* (4 ed.). Oslo: Gyldendal akademisk.

Braun, V., & Clarke, V. (2013). *Successful qualitative research: A practical guide for beginners.* London: Sage.

Bryman, A. (2016). *Social research methods.* New York: Oxford University Press.

Carson, J. B., Tesluk, P. E., & Marrone, J. A. (2007). Shared leadership in teams: An investigation of antecedent conditions and performance. *Academy of Management Journal,* 50(5), 1217-1234.

Cherns, A. (1976). The principles of sociotechnical design. *Human Relations,* 29(8), 783-792.

Cockburn, A., & Highsmith, J. (2001). Agile software development, the people factor. *Computer,* 34(11), 131-133.

Cummings, T. G. (1978). Self-regulating work groups: A socio-technical synthesis. *Academy of Management Review*, 3(3), 625-634.

Cummings, T. G., & Worley, C. G. (2015). *Organization development and change.* Stamford: Cengage Learning.

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.

Druskat, V. U., & Wheeler, J. V. (2004). How to lead a self-managing team. *IEEE Engineering Management Review,* 32(4), 21-28.

Dybå, T. (2000). Improvisation in small software organizations. *IEEE Software*, 17(5), 82-87.

Emery, F. E., & Thorsrud, E. (1976). *Democracy at work: The report of the Norwegian industrial democracy program*. London: Springer.

Emery, F. E., & Trist, E. L. (1972). *Towards a social ecology.* London: Plenum Press.

Evans, W. R., & Davis, W. D. (2005). High-performance work systems and organizational performance: The mediating role of internal social structure. *Journal of Management,* 31(5), 758-775.

Gerwin, D., & Moffat, L. (1997). Authorizing processes changing team autonomy during new product development. *Journal of Engineering and Technology Management*, 14(3-4), 291-313.

Gonçalves, L., & Linders, B. (2013). *Getting value out of Agile retrospectives: a toolbox of retrospective exercises.* Morrisville, NC: Lulu.com.

Goodman, P. S., Devadas, R., & Hughson, T. L.G. (1988). Groups and Productivity; Analyzing the Effectiveness of Self-Managing Teams. In J. P. Campbell & C. R. J (Eds.), *Productivity in organizations* (pp. 295-327). San Francisco: Jossey-Bass.

Gren, L. (2017). *Psychological Group Processes when Building Agile Software Development Teams.* (Ph.D. thesis), University of Gothenburg, Gothenburg, Sweden.

Guba, E. G., & Lincoln, Y. S. (1994). Competing paradigms in qualitative research. In N. K. Denzin & Y. S. Lincoln (Eds.), *Handbook of qualitative research* (pp. 105-117). Thousand Oaks, CA: Sage.

Hackman, J. R. (1987). The design of work teams. In J. W. Lorsch (Ed.), *Handbook of organizational behavior* (pp. 315-342). Englewood Cliffs, NJ: Prentice Hall.

Hackman, J. R., & Oldham, G. R. (1980). *Work redesign*. Reading: Addison-Wesley.

Herbst, P. G. (1974). *Sociotechnical design.* London: Tavistock.

Herzberg, F. I. (1966). *Work and the nature of man*. Oxford, England: World.

Hoegl, M., & Parboteeah, P. (2006). Autonomy and teamwork in innovative projects. *Human Resource Management*, 45(1), 67-79.

Kakar, A. K. (2014). Teaching theories underlying agile methods in a systems development course. Paper presented at the *System Sciences (HICSS), 2014 47th Hawaii International Conference*, Waikoloa, Hawaii.

Katzenbach, J. R., & Smith, D. K. (1993). *The discipline of teams*. Boston, MA: Harvard Business Press.

Kniberg, H., & Skarin, M. (2010). *Kanban and Scrum-making the most of both.* Morrisville, NC: Lulu.com.

Langfred, C. W. (2000). The paradox of self-management: Individual and group autonomy in work groups. *Journal of Organizational Behaviour*, 21(5), 563-585.

Langfred, C. W. (2005). Autonomy and performance in teams: The multilevel moderating effect of task interdependence. *Journal of Management*, 31(4), 513-529.

Langfred, C. W. (2007). The downside of self-management: A longitudinal study of the effects tf conflict on trust, autonomy, and task interdependence in self-managing teams. *Academy of Management Journal,* 50(4), 885-900.

Latham, G. P., & Saari, L. M. (1979). The Effects of Holding Goal Difficulty Constant on Assigned and Participatively Set Goals. *Academy of Management Journal,* 22(1), 163-168.

Lawler, E. E. (1992). *The ultimate advantage: Creating the high-involvement organization.* San Francisco: Jossey-Bass.

Lincoln, Y. S., & Guba, E. G. (1985). *Naturalistic inquiry.* Newbury Park, CA: Sage.

Manz, C. C., & Sims Jr, H. P. (1982). The potential for "groupthink" in autonomous work groups. *Human Relations*, 35(9), 773-784.

Manz, C. C., & Sims Jr, H. P. (1987). Leading workers to lead themselves: The external leadership of self-managing work teams. *Administrative Science Quarterly*, 32(1), 106-129.

Mathieu, J. E., Gilson, L. L., & Ruddy, T. M. (2006). Empowerment and team effectiveness: an empirical test of an integrated model. *Journal of Applied Psychology,* 91(1), 97-108.

Moe, N. B., Aurum, A., & Dybå, T. (2012). Challenges of shared decision-making: A multiple case study of agile software development. *Information and Software Technology,* 54(8), 853-865.

Moe, N. B., Dingsøyr, T., & Dybå, T. (2008). Understanding self-organizing teams in agile software development. Paper presented at the *ASWEC 2008. Proceedings of the 19th Australian Conferences on Software Engineering*, Perth, Australia.

Moe, N. B., Dingsøyr, T., & Dybå, T. (2009). Overcoming barriers to self-management in software teams. *IEEE Software*, 26(6), 20-26.

Morgan, G. (2006). *Images of organization.* London: Sage.

Nerur, S., & Balijepally, V. (2007). Theoretical reflections on agile development methodologies. *Communications of the ACM*, 50(3), 79-83.

Nerur, S., Cannon, A., Balijepally, V., & Bond, P. (2010). Towards an understanding of the conceptual underpinnings of agile development methodologies. In Dingsøyr T, Dybå T, & M. N (Eds.), *Agile Software Development* (pp. 15-29). Berlin: Springer.

Nickerson, R. S. (1998). Confirmation bias: A ubiquitous phenomenon in many guises. *Review of General Psychology*, 2(2), 175-220.

Nonaka, I. (1994). A Dynamic Theory of Organizational Knowledge Creation. *Organization Science*, 5(1), 14-37.

Parker, S., & Wall, T. D. (1998). *Job and work design: Organizing work to promote well-being and effectiveness*. Thousand Oaks, CA: Sage.

Pearce III, J. A., & Ravlin, E. C. (1987). The design and activation of self-regulating work groups. *Human Relations*, 40(11), 751-782.

Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., & Still, J. (2008). The impact of agile practices on communication in software development. *Empirical Software Engineering*, 13(3), 303-337.

Ratliff, R. L., Beckstead, S. M., & Hanks, S. H. (1999). The use and management of teams: A how-to guide. *Quality Progress*, 32(6), 31-38.

Schick, A. G., Gordon, L. A., & Haka, S. (1990). Information overload: A temporal approach. *Accounting, Organizations and Society*, 15(3), 199-220.

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

Stewart, G. L., & Manz, C. C. (1995). Leadership for self-managing work teams: A typology and integrative model. *Human Relations*, 48(7), 747-770.

Stray, V. G., Moe, N. B., & Dingsøyr, T. (2011). Challenges to teamwork: a multiple case study of two agile teams. In A. Sillitti, O. Hazzan, E. Bache, & X. Albaladejo (Eds.), *Agile Processes in Software Engineering and Extreme Programming* (pp. 146-161). Berlin: Springer.

Stray, V., Sjøberg, D. I., & Dybå, T. (2016). The daily stand-up meeting: A grounded theory study. *Journal of Systems and Software*, 114, 101-124.

Susman, G. I. (1976). *Autonomy at work: A sociotechnical analysis of participative management*. New York: Praeger.

Taggar, S., Hackew, R., & Saha, S. (1999). Leadership emergence in autonomous work teams: Antecedents and outcomes. *Personnel Psychology,* 52(4), 899-926.

Tata, J., & Prasad, S. (2004). Team self-management, organizational structure, and judgments of team effectiveness. *Journal of Managerial Issues,* 16(2), 248-265.

Tjora, A. (2017). *Kvalitative forskningsmetoder i praksis.* Oslo: Gyldendal akademisk.

Trist, E. L. (1981). The sociotechnical perspective: The evolution of sociotechnical systems as a conceptual framework and as an action research program. In A. H. Van der Ven & W. F. Joyce (Eds.), *Perspective on Organization Design and Behavior* (pp. 19-75). New York: John Wiley and Sons.

Uhl-Bien, M., & Graen, G. B. (1992). Self-management and team-making in cross-functional work teams: Discovering the keys to becoming an integrated team. *The Journal of High Technology Management Research*, 3(2), 225-241.

Van Amelsvoort, P. (2000). *The Design of Work and Organisation: the Modern Sociotechnical Systems approach* (1 ed.). Vlijmen: ST-GROEP.

Van der Vegt, G. S., Bunderson, S., & Kuipers, B. (2010). Why turnover matters in self-managing work teams: Learning, social integration, and task flexibility. *Journal of Management,* 36(5), 1168-1191.

Van Eijnatten, F. M. (1993). *The paradigm that changed the work place.* Assen: Van Gorcum.

Van Mierlo, H., Rutte, C. v., Vermunt, J., Kompier, M., & Doorewaard, J. (2006). Individual autonomy in work teams: The role of team autonomy, self-efficacy, and social support. *European Journal of Work and Organizational Psychology*, 15(3), 281-299.

Wageman, R. (1997). Critical success factors for creating superb self-managing teams. *Organizational Dynamics*, 26(1), 49-61.

Wageman, R., & Fisher, C. (2014). Who's in charge here? The team leadership implications of authority. In D. V. Day (Ed.), *The Oxford handbook of leadership and organizations* (pp. 455-481). New York: Oxford University Press.

Watson, T. J. (2008). *Sociology, Work and Industry.* London: Routledge.

Yin, R. K. (2014). *Case study research: Design and methods* (5 ed.). Thousand Oaks, CA: Sage.

9 Appendices

Appendix A: Interview Guide - Formal Leadership Responsibilities

Introduksjon

- Takke personen for å stille til intervju
    - Fortelle kort om gangen i intervjuet, informere om varighet ca 1 time
- Presentere oss
    - Navn, studie, masteroppgave om autonome team, del av A-team i SINTEF
- Bekrefte konfidensialitet og anonymitet
    - Gi dem samtykkeerklæringen
- Spørre om tillatelse til å ta opptak av intervjuet (sette på opptaker)
- Informere om at intervjuet transkriberes av oss og behandles som annet datamateriale i A-teamprosjektet

Bli kjent

- Kan du fortelle litt om jobben/stillingen din?
    - Hvor lenge har du jobbet for selskapet?
    - Hva er og hvor lenge har du hatt din nåværende stilling?
    - Hva innebærer stillingen din?
- Hva er din utdannelsesbakgrunn?
    - Har du noen annen erfaring/kurs som er relevant for stillingen din?

Teamet

- Kan du beskrive teamet/(-ene) du jobber med?
    - Hvor mange team?
    - Størrelse på teamet?
    - Hva gjør teamet?
    - Hvem/hvilke roller består teamet av?
        - Er rollene definerte eller ikke definerte?
        - Hvordan fordeles oppgaver i teamet?
    - Hvordan setter teamet mål?
        - Hvordan sikrer man en felles retning?
    - Hvordan tar teamet beslutninger?

    - Hva tenker du om måten teamet er satt sammen på (med tanke på oppgavene som skal løses)?
    - Hva gjør teamet for å forbedre prosessene sine?
- Hvilken rolle har du overfor teamene?
    - Hvilke oppgaver og ansvarsområder har du overfor teamet?
    - Hva er funksjonen/formålet med stillingen din?
    - Hvorfor tror du bedriften har “din rolle” overfor teamene?
    - Ser du på deg selv som en del av teamet?
- Hvordan samhandler du med teamet?
    - Hvordan foregår kommunikasjon og kontakt med teamet?

Bidrag til teamet

- Hva ser du på som ditt viktigste bidrag til teamet?
- Hvilke utfordringer har du i jobben din? (i utførelsen av dine oppgaver)
- Hvilken betydning har bakgrunnen din for jobben du gjør?
    - Noe annen erfaring/utdanning du skulle ønske du hadde?
    - Kan du nevne et eksempel på en situasjon der du følte din bakgrunn var spesielt nyttig?

Episoder om rollen

- Nevn en teknisk utfordring teamet har stått overfor?
    - Hva var din rolle i dette?
    - Var det noen andre som hadde ansvar i forbindelse med dette?
- Nevn en mellommenneskelig utfordring teamet har stått overfor?
    - Hva var din rolle i dette?
    - Var det noen andre som hadde ansvar i forbindelse med dette?

Avslutning

- Er det noe du vil legge til som vi ikke allerede har snakket om?
- Har du noen spørsmål til oss?
- Fortelle at datamaterialet vil transkriberes og anonymiseres. Lagres og slettes etter A-teams retningslinjer. Du kan når som helst trekke deg fra studien vår, og i så fall vil datamaterialet slettes.
- Takke for intervjuet
    - Vise til samtykkeerklæringen med vår kontaktinfo
    - Er det greit om vi tar kontakt med deg igjen dersom vi har flere spørsmål?

Appendix B: Interview Guide - No Formal Leadership Responsibilities

Introduksjon

- Takke personen for å stille til intervju
    - Fortelle kort om gangen i intervjuet, informere om varighet ca 1 time
- Presentere oss
    - Navn, studie, masteroppgave om autonome team, del av A-team i SINTEF
- Bekrefte konfidensialitet og anonymitet
    - Gi dem samtykkeerklæringen
- Spørre om tillatelse til å ta opptak av intervjuet (sette på opptaker)
- Informere om at intervjuet transkriberes av oss og behandles som annet datamateriale i A-teamprosjektet

Bli kjent

- Kan du fortelle litt om jobben/stillingen din?
    - Hvor lenge har du jobbet for selskapet?
    - Hva er og hvor lenge har du hatt din nåværende stilling?
    - Hva innebærer stillingen din?

Teamet

- Kan du beskrive teamet du jobber i?
    - Størrelse på teamet?
    - Tegn opp teamet (tegning med rollene det består av og tilknytning til linja)
    - Hva gjør teamet?
    - Hvem/hvilke roller består teamet av?
        - Er rollene definerte eller ikke definerte?
        - Hva tenker du om måten teamet er satt sammen på (med tanke på oppgavene som skal løses)?
        - Hvordan fordeles oppgaver i teamet?
- Hvilke utfordringer har du i jobben din? (i utførelsen av dine oppgaver)
- Hvordan setter teamet mål?
        - Hvordan sikrer man en felles retning?
- Hvordan tar teamet beslutninger?
- Hva gjør teamet for å forbedre prosessene sine?
- Hva legger du i begrepet autonomt/selvstyrt? Er teamet

autonomt/selvstyrt?

- Hvilken rolle har team lead og tech lead i ditt teamene?
    - Hvilke oppgaver og ansvarsområder har disse rollene?
    - Hva er funksjonen/formålet med disse rollene?
    - Hvorfor tror du bedriften har team lead/tech lead?

Episoder om rollen

- Nevn en teknisk utfordring teamet har stått overfor?
    - Hva var din rolle i dette?
    - Hva gjorde de med lederansvar?
- Nevn en mellommenneskelig utfordring teamet har stått overfor?
    - Hva var din rolle i dette?
    - Hva gjorde de med lederansvar?

Avslutning

- Er det noe du vil legge til som vi ikke allerede har snakket om?
- Har du noen spørsmål til oss?
- Fortelle at datamaterialet vil transkriberes og anonymiseres. Lagres og slettes etter A-teams retningslinjer. Du kan når som helst trekke deg fra studien vår, og i så fall vil datamaterialet slettes.
- Takke for intervjuet
    - Vise til samtykkeerklæringen med vår kontaktinfo
    - Er det greit om vi tar kontakt med deg igjen dersom vi har flere spørsmål?

Appendix C: Interview Guide - Follow-up Interviews

Introduksjon

- Takke for sist
- Kort oppsummering av hva vi har gjort og hvordan det går, hvor vi er på vei
- Noe du har tenkt på siden sist? Noe du ikke fikk sagt?

Hoveddel

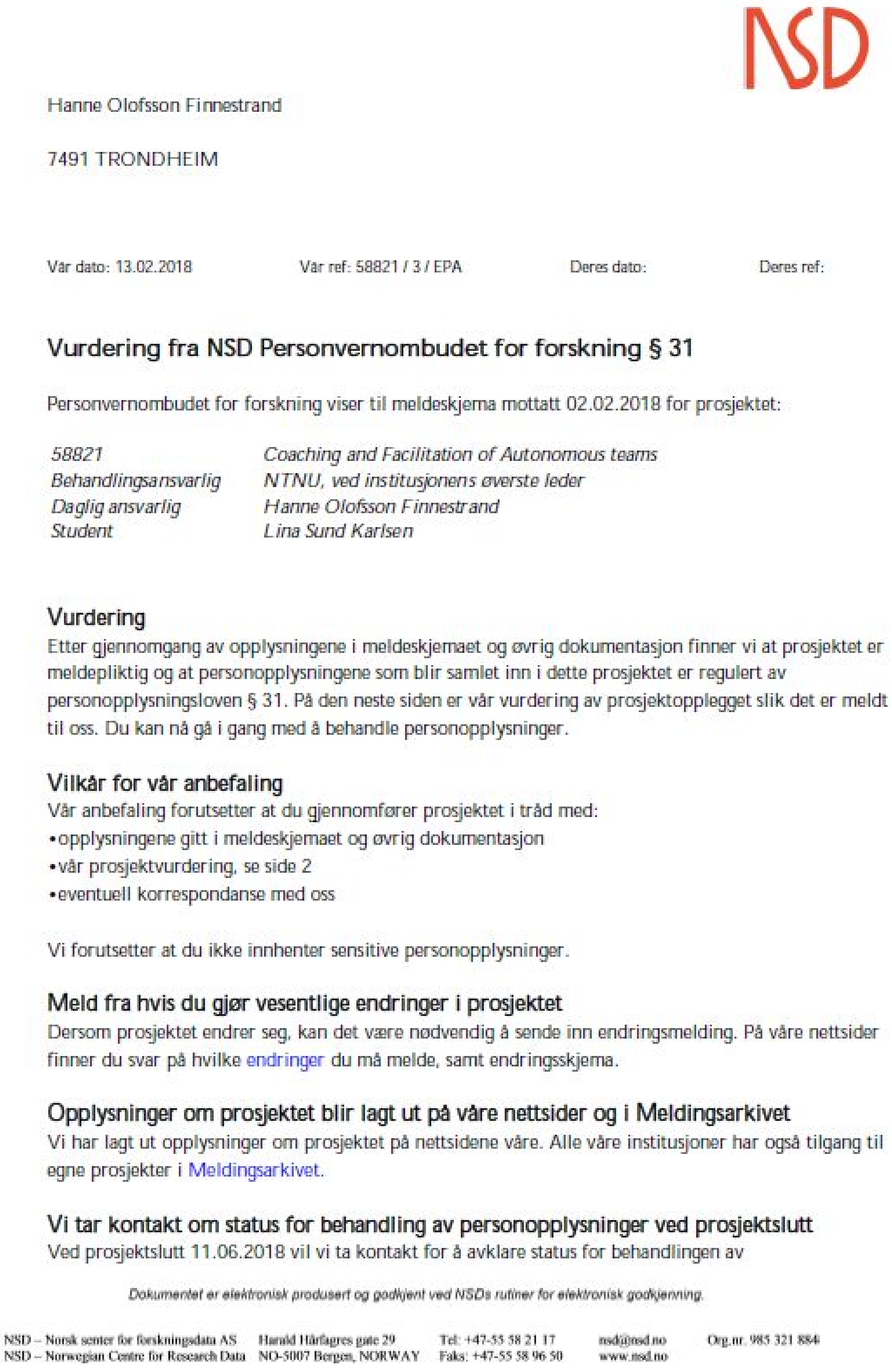
- Hvordan går det i teamet nå?
    - Hva jobber dere med? Travelt? Utfordringer? Fremgang?
- Tegn opp teamet (tegning med rollene det består av og tilknytning til linja)
- Hvem rapporterer teamet til? Hvem i organisasjonen følger opp teamet?
    - På hvilken måte?
- Da du ble team lead, hva slags opplæring og hjelp fikk du?
- Hva legger du i begrepet autonomt/selvstyrt? Er teamet autonomt/selvstyrt?
    - Hva tenker du om din rolle med tanke på å oppnå autonomi?
    - Hvilke av oppgavene du gjør tenker du må ligge hos team lead?
    - Har dere hatt noen utfordringer/hindringer på veien mot mer autonomi?

- **Spesifikt for dette objektet, ta opp noe fra sist dersom det er interessant**

Avslutning

- Er det noe du vil legge til som vi ikke har snakket om?
- Har du noen spørsmål til oss?
- Takke for intervjuet, fortelle at datamateriale behandles konfidensielt etter A-teams retningslinjer

Appendix D: Approval Norwegian Centre for Research Data



NSD

Hanne Olofsson Finnestrand

7491 TRONDHEIM

Vår dato: 13.02.2018   Vår ref: 58821 / 3 / EPA   Deres dato: Deres ref:

Vurdering fra NSD Personvernombudet for forskning § 31

Personvernombudet for forskning viser til meldeskjema mottatt 02.02.2018 for prosjektet:

58821	*Coaching and Facilitation of Autonomous teams*
Behandlingsansvarlig	*NTNU, ved institusjonens øverste leder*
Daglig ansvarlig	*Hanne Olofsson Finnestrand*
Student	*Lina Sund Karlsen*

Vurdering

Etter gjennomgang av opplysningene i meldeskjemaet og øvrig dokumentasjon finner vi at prosjektet er meldepliktig og at personopplysningene som blir samlet inn i dette prosjektet er regulert av personopplysningsloven § 31. På den neste siden er vår vurdering av prosjektopplegget slik det er meldt til oss. Du kan nå gå i gang med å behandle personopplysninger.

Vilkår for vår anbefaling

Vår anbefaling forutsetter at du gjennomfører prosjektet i tråd med:

- opplysningene gitt i meldeskjemaet og øvrig dokumentasjon
- vår prosjektvurdering, se side 2
- eventuell korrespondanse med oss

Vi forutsetter at du ikke innhenter sensitive personopplysninger.

Meld fra hvis du gjør vesentlige endringer i prosjektet

Dersom prosjektet endrer seg, kan det være nødvendig å sende inn endringsmelding. På våre nettsider finner du svar på hvilke endringer du må melde, samt endringsskjema.

Opplysninger om prosjektet blir lagt ut på våre nettsider og i Meldingsarkivet

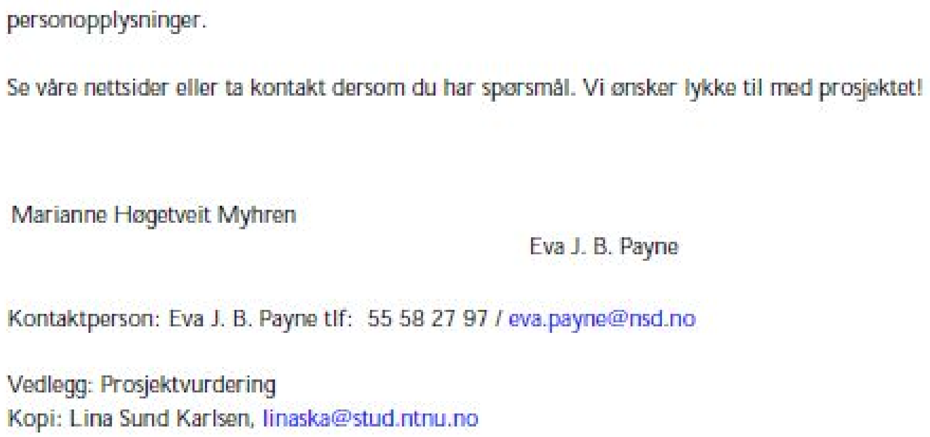
Vi har lagt ut opplysninger om prosjektet på nettsidene våre. Alle våre institusjoner har også tilgang til egne prosjekter i Meldingsarkivet.

Vi tar kontakt om status for behandling av personopplysninger ved prosjektslutt

Ved prosjektslutt 11.06.2018 vil vi ta kontakt for å avklare status for behandlingen av

Dokumentet er elektronisk produsert og godkjent ved NSDs rutiner for elektronisk godkjenning.

NSD – Norsk senter for forskningsdata AS   Harald Hårfagres gate 29   Tel: +47-55 58 21 17   nsd@nsd.no   Org.nr. 985 321 884
NSD – Norwegian Centre for Research Data   NO-5007 Bergen, NORWAY   Faks: +47-55 58 96 50   www.nsd.no



personopplysninger.

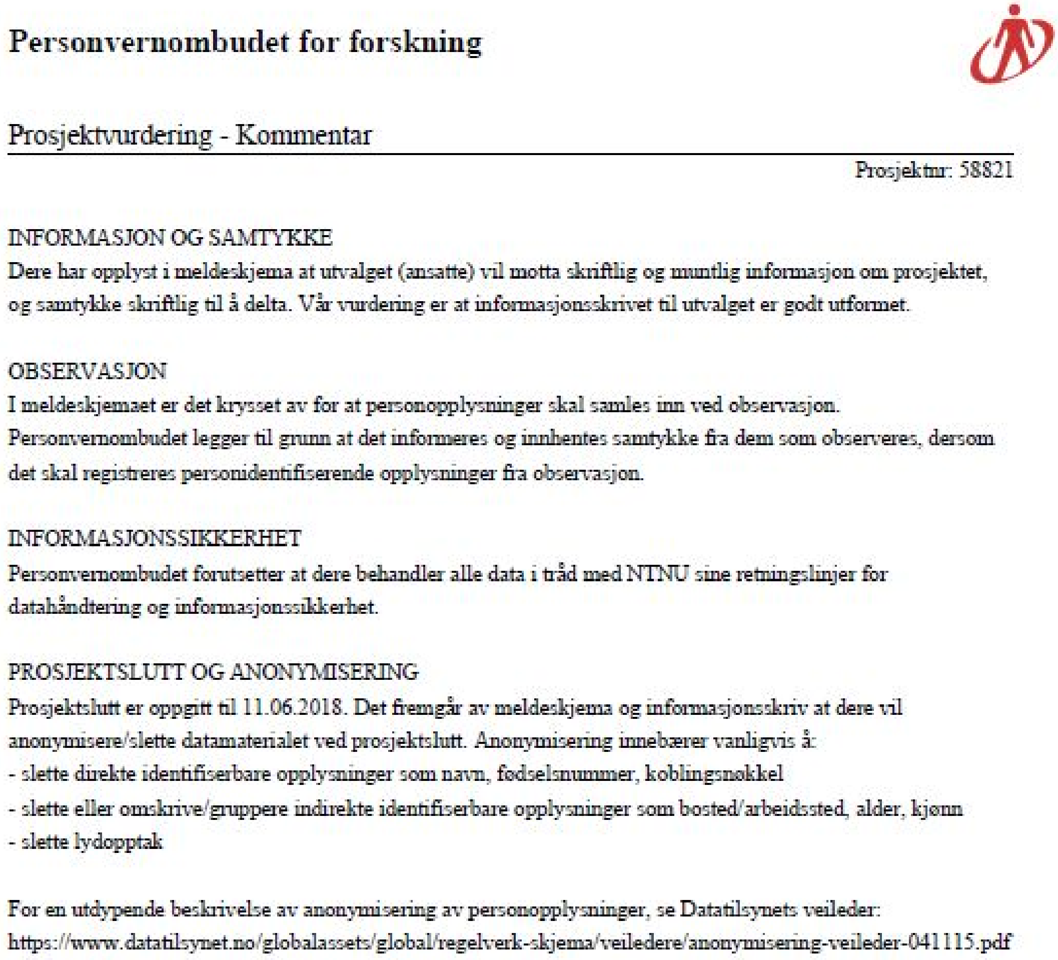
Se våre nettsider eller ta kontakt dersom du har spørsmål. Vi ønsker lykke til med prosjektet!

Marianne Høgetveit Myhren

Eva J. B. Payne

Kontaktperson: Eva J. B. Payne tlf: 55 58 27 97 / eva.payne@nsd.no

Vedlegg: Prosjektvurdering
Kopi: Lina Sund Karlsen, linaska@stud.ntnu.no



Personvernombudet for forskning

Prosjektvurdering - Kommentar

Prosjektnr: 58821

INFORMASJON OG SAMTYKKE
Dere har opplyst i meldeskjema at utvalget (ansatte) vil motta skriftlig og muntlig informasjon om prosjektet, og samtykke skriftlig til å delta. Vår vurdering er at informasjonsskrivet til utvalget er godt utformet.

OBSERVASJON
I meldeskjemaet er det krysset av for at personopplysninger skal samles inn ved observasjon. Personvernombudet legger til grunn at det informeres og innhentes samtykke fra dem som observeres, dersom det skal registreres personidentifiserende opplysninger fra observasjon.

INFORMASJONSSIKKERHET
Personvernombudet forutsetter at dere behandler alle data i tråd med NTNU sine retningslinjer for datahåndtering og informasjonssikkerhet.

PROSJEKTSLUTT OG ANONYMISERING
Prosjektslutt er oppgitt til 11.06.2018. Det fremgår av meldeskjema og informasjonsskriv at dere vil anonymisere/slette datamaterialet ved prosjektslutt. Anonymisering innebærer vanligvis å:
- slette direkte identifiserbare opplysninger som navn, fødselsnummer, koblingsnøkkel
- slette eller omskrive/gruppere indirekte identifiserbare opplysninger som bosted/arbeidssted, alder, kjønn
- slette lydopptak

For en utdypende beskrivelse av anonymisering av personopplysninger, se Datatilsynets veileder: https://www.datatilsynet.no/globalassets/global/regelverk-skjema/veiledere/anonymisering-veileder-041115.pdf

Appendix E: Consent Form for Participation in the Research Project

Samtykkeerklæring for deltagelse i forskningsprosjekt

”Masteroppgave om autonome team”

Vi er tre studenter; Bjørn Dahl, Stine Schjødt-Osmo og Lina Sund Karlsen, som skriver masteroppgave ved institutt for industriell økonomi og teknologiledelse ved NTNU. Masteroppgaven har tilrettelegging for autonome team som tema. For å studere dette vil vi intervjue personer i ulike stillinger både i og utenfor autonome team.

Deltakelse i prosjektet vil innebære intervju med varighet på omtrent en time. Spørsmålene vil hovedsakelig dreie seg om tilrettelegging for autonome team, og informantens deltagelse og erfaringer knyttet til dette. Intervjuet vil bli tatt opp og transkribert av oss tre studentene, og vi kommer til å ta notater underveis i intervjuet.

Vi er underlagt taushetsplikt, og alle personopplysninger vil bli behandlet konfidensielt. Alt datamateriale lagres i tråd med SINTEFs avtale med bedriften. All data vil anonymiseres, og koblingsnøkkel lagres adskilt fra øvrige data for å sikre konfidensialitet. Enkeltpersoner vil anonymiseres i publikasjonen, slik at deltakere ikke vil kunne gjenkjennes. Vår masteroppgave skal etter planen avsluttes juni 2018, og alt datamateriale vil da slettes. Informanter som ønsker det kan få en kopi av oppgaven etter at den er levert.

Prosjektet gjennomføres i samarbeid med SINTEFs pågående prosjekt om autonome team (A-team), og vi har derfor veiledere fra både NTNU og SINTEF. Veileder fra NTNU er Hanne Olofsson Finnestrand, som kan kan kontaktes på e-post: hanne.finnestrand@ntnu.no. Veileder fra SINTEF er Nils Brede Moe, som kan kontaktes på e-post: Nils.B.Moe@sintef.no. Vi kan kontaktes på følgende e-poster: bjornhdahl@gmail.com, stine.schjodt.osmo@gmail.com og linaska@stud.ntnu.no.

Det er frivillig å delta i studien, og du kan når som helst trekke ditt samtykke uten å oppgi noen grunn. Dersom du trekker deg, vil alle opplysninger om deg bli slettet.

Jeg samtykker herved til å delta i intervju

(Signert av prosjektdeltaker, dato)