AALBORG
UNIVERSITY

STUDENT REPORT

How to Motivate and Control Software Developers Working From Home During the COVID-19 Pandemic

Master's Thesis
IT Design and Application Development

Amal Lahdjel

Hanan Mohamoud Ahmed

Lily Thi Kimie Huynh

Louis Nikolaj Rasmussen

Supervised by
John Stouby Persson
June 11th, 2021

Abstract

Context. As software developers were forced to work from home due to COVID-19 pandemic, managers had to rethink how to control and motivate software developers. *Objective.* This study investigates how managers can motivate and control software developers who are working from home during the pandemic. *Method.* We used a case study approach to investigate a Scrum team from a Danish software organization that used Scaled Agile Framework (SAFe). In order to get an in-depth understanding of the participants involved, semi-structured interviews were conducted remotely and subsequently transcribed. In addition, qualitative content analysis was used to analyze the data, where Kirsch's control modes were applied to determine what kind of controls the organization enacted and Herzberg's motivation-hygiene theory to find out if the software developers were dissatisfied or satisfied with the controls. However, as we argue that the relationship between control and motivation is transactional, we also investigate how the software developers' motivation influences the control. *Results.* We found that the identified control enactments at the organization had an effect on the software developers' motivation, however, it was also evident that their motivation had an effect on the controls. *Conclusion.* The majority of these controls had a positive impact, which suggests that the SAFe worked well in the context of working from home during the COVID-19 pandemic. In addition, it is important for managers to be aware of the transactional relationship when controlling and motivating software developers who are working from home.

Table of contents

1. Introduction

The infectious coronavirus disease (COVID-19) was first learned by the World Health Organization (WHO) in December 2019 through a report of cases of viral pneumonia in Wuhan, China (WHO, 2020a). When infected with the COVID-19 virus, the majority of people experienced mild to moderate respiratory illness, and in some cases developed serious symptoms (WHO, 2021). The COVID-19 outbreak spread rapidly internationally, which resulted in WHO announcing it as a pandemic on March 11th, 2020. Many countries, including Denmark, declared states of emergency or lockdowns (WHO, 2020b; Kaplan, Frias & McFall-Johnsen, 2020). In order to hinder the spread of the virus, the Danish government implemented initiatives, such as public employees, who did not perform critical functions, had to work from home. Private employers were also strongly encouraged to ensure that their employees, as many as possible, were working from home or taking time off (Regeringen, 2020). This resulted in technology companies closing their offices, as they did not perform critical functions, thus employees, who normally work in an office setting, had to switch to working from home (Ford et al., 2020, p. 1). Working from home during a pandemic is not the same as working from home during normal circumstances. Rather than working in a well-appointed home office, some people are working in their bedrooms, at kitchen tables, and on sofas while being distracted by children, parents, roommates, etc. Parents juggle work with childcare and homeschooling, as school and child care centers are closed (Ralph et al., 2020, p. 2). Thus, working from home during a pandemic can have an effect on employees' motivation, which we also found in our previous study in the 9th semester. We found, among other things, that software developers had difficulties in communicating with each other (Lahdjel, Mohamoud, Rasmussen, Huynh & Untaru, 2021, p. 49), and had their working hours extended (Lahdjel et al., 2021, p. 45). Work motivation is thus an important matter, as according to Kanfer, Chen and Pritchard, an individual's work motivation reflects the opportunity for improving the productivity in the organization (Kanfer, Chen & Pritchard, 2008, p. 2). Furthermore, employees who are motivated may be more willing to apply higher levels of effort to their job, which leads to an increase in performance and decrease in absenteeism and turnover (França et al., 2013, p. 9).

For a long time, control in organizations has been a topic of interest for researchers and practitioners, who recognize that control enactments are needed for organizations to achieve their goals (Kirsch, 1996, p. 1). For information systems development (ISD) projects to not exceed over time and over budget, the exercise of control is needed (Huang Chua & Myers, 2018, p. 173). Thus, control is seen as a powerful managerial instrument to increase the success rate of ISD projects (Remus, Wiener, Saunders, Mähring & Kofler, 2016, p. 2). With the use of control, ISD managers (controllers) can steer the project participants' (controllees) behaviors to evoke their contributions and cooperation, and by that maintain the progress to reach the project goals (Remus et al., 2016, p. 2). According to Schrage, employees who work from home are limited to interacting with each other as well as their managers (Schrage, 2021, p. 88). This suggests that the management has a lack of control over the employees, which is problematic, as managers still need to ensure that projects are on track, and that they will meet the customers' needs (Schrage, 2021, p. 88).

Examining how managers can both motivate and control software developers while they work from home during the COVID-19 pandemic, is therefore relevant. Thus, the research question which the group will investigate is:

How can managers motivate and control software developers who are working from home during the COVID-19 pandemic?

In this study, we focus on the relationship between control and motivation, which Elkjaer and Simpson call; *transaction*. In a transaction, the components are subject to change, and their character affects and is affected by the transaction (Elkjaer & Simpson, 2011, p. 69). In other words, we view control and motivation as being subject to change, and that they reciprocally affect and influence each other. When analyzing both control enactments and motivation of the software developers, we will thus interpret how these can relate and influence each other. By addressing the research question, we hope to contribute with suggestions on how managers can both motivate and control software developers at the specific organization under study during the pandemic. In regard to research, we found that studies on control in the context of a pandemic are scarce, whereas studies on specifically motivation are non-existing. This is further elaborated on in the next chapter. Regarding practice, working from home has affected a lot of people, both negatively and positively. Depending on how it went for the different organizations, they might be more acceptable for having their employees working more from home when the pandemic is over, thus potentially changing how we will work. This is supported by Bartik, Cullen, Glaeser, Luca, and Stanton, as their work suggested that remote work is likely to remain at higher levels after the pandemic, and that the pandemic may have a persistent impact on the nature of work (Bartik, Cullen, Glaeser, Luca & Stanton, 2020, p. 4). Our results may thus be of help to managers by offering some suggestions, which they can consider in order to motivate and control software developers while they work from home. In our study, we follow the definition of control by Remus et al., who define it as: "*any attempt to ensure that individuals working on a project act in a manner consistent with organizational goals*'' (Remus et al., 2016, p. 3). Since a typical control situation is where there is a controller, who exercises control over a controllee, or a group of controllees (Remus et al., 2016, p. 3), we view managers or any other positions with certain decision-making responsibilities as the controllers, and software developers as the controllees. Managers are defined as the ones who use control to steer the behavior of the project participants, which are the software developers in our case. In regard to motivation in our study, we take the definition by Kanfer et. al, who define it as; *“a psychological process that influences how personal effort and resources are allocated to actions pertaining to work, including the direction, intensity, and persistence of these actions''* (Kanfer et al., 2008, p. 5).

2. Research Background

This chapter presents literature within and outside ISD on control and motivation, and the relationship between them, in ISD and the research gaps within these.

A research background is included in this study to argue for the relevance of the research question. By including a research background, we can present what has already been studied in the field of control and motivation in and outside ISD and find current research gaps. In addition, we will argue for the transactional relationship between control and motivation by looking into existing research about how they affect and influence each other. At the end of this chapter, we will arrive at a research question based on these research gaps.

2.1 Literature on Control in ISD

Within the literature on control in ISD, multiple studies can be found. To mention a few, Cram and Wiener investigated how control activities relate to perception of control legitimacy. Control legitimacy refers to the controller's perception of the enacted control enactment, whether it gives justice, autonomy, group identification, and competence development (Cram & Wiener, 2018, p. 713). They found that managers, who consider employees' desires when enacting controls, can encourage improved organizational outcomes (Cram & Wiener, 2018, pp. 731-732). Managers, who are aware of these desires, can enhance individual, group, and organizational performance through increasingly motivated and engaged employees, and avoid employee dissatisfaction and turnover intentions (Cram & Wiener, 2018, p. 732). In addition, Huang Chua and Meyers have investigated how social context has an influence on control in ISD (Huang Chua & Meyers, 2018, p. 174) and found, among other things, that control enactment also involves the controllees' interpretation (Huang Chua & Meyers, 2018, p. 183). Controllees can interpret the controls wrongly or respond in a way that changes the efficacy of controls (Huang Chua & Meyers, 2018, p. 186). Thus, they suggested that both the controller and controllee should negotiate control, rather than the controller enacting control unilaterally (Huang Chua & Meyers, 2018, p. 186).

What the aforementioned studies have in common is that they both make use of the control modes theory in their research on control. This theory was used to better understand and explain control in information systems operations and development (Huang Chua & Meyer, 2018, p. 173; Cram & Wiener, 2018, p. 715), as it refers to the overall characteristics of the controls (Cram & Wiener, 2018, p. 715). In addition, they also take the perspective of the controllee into consideration in their studies (Huang Chua & Meyers, 2018, p. 177; Cram & Wiener, 2018, p. 712).

According to Cram and Wiener, only a few studies on ISD control have collected data from the perspective of both the controller and controllee, thus the research in ISD control still remains focused on the controller (Cram & Wiener, 2018, p. 715). The lack of focus on the controllee can therefore be considered to be a research gap within the ISD control. In addition, how control is enacted during a pandemic is scarce. Among the few studies that have investigated control within this context is the paper by Schrage, who investigated how software development teams can be controlled during the COVID-19 pandemic by using control enactments used in agile or traditional software development teams, though only through a literature review (Schrage, 2021, pp. 87-88). However, studies that regard ISD control during the COVID-19 pandemic were not included in his study, even though these were of particular interest (Schrage, 2021, p. 91). The fact that these were not included, suggests that studies regarding ISD control during the COVID-19 pandemic are scarce. This might be due to the fact that the pandemic is relatively recent and still ongoing. Therefore, it is still considered a research gap that should be investigated. During our study, however, we were able to find a study by Schmidtner, Doering, and Timinger (2021) which argued that the implementation of an agile method was useful, while working from home during the pandemic. They investigated how the agile method was affected by the pandemic and saw that the agility made it easier for the organizations to adapt to the changes caused by the pandemic (Schmidtner, Doering & Timinger, 2021, p. 10).

2.2 Literature on Motivation in ISD

The topic of motivation in ISD can also be found in existing studies. Baddoo, Hall, and Jagielska investigated how motivation has an effect on software development work in a high maturity company (Baddoo, Nathan & Hall, 2006, p. 219). The term “maturity” relates to how capable an organization is of achieving continuous improvement, thus having high maturity means the company is highly optimized and competitive (Sigal, 2016). The results showed that the biggest motivators to good performance in a high maturity organization are pay and benefits, recognition, and opportunities for achievement (Baddoo et al., 2006 p. 227). Other things such as challenging work and job security closely followed these. Surprisingly, factors like autonomy, sense of ownership and increased responsibility were less represented (Baddoo et al., 2006, p. 226). However, another paper by Beecham, Baddoo, Hall, Robinson, and Sharp found contradicting results, as they examined what motivates and demotivates developers, and how existing models addressed motivation. The study found, through a collection of literature, that the most frequently cited motivators for software engineers are ‘the need to identify with the task’, such as having clear goals, a personal interest, understanding the purpose of the task, and how it fits with the whole (Beecham, Baddoo, Hall, Robinson & Sharp, 2008, p. 874). Less effective motivators were ‘good management’, consisting of team-building and good communication, and ‘career path’, where the employees have opportunity for advancement and career planning (Beecham et al., 2008, p. 868). The findings also indicated that motivated engineers will increase their productivity and remain longer in the organization, whereas if demotivated, they may leave the organization or take more sick-leaves (Beecham et al., 2008, p. 860). The contradicting findings may suggest that the motivation of software developers can vary depending on their individual context. For example, the role of pay and benefits, as a strong motivator for good performance, could be a reflection of the economic environment of the time and the state of the computer industry as a whole. As of this moment, the literature that investigates motivation in ISD in the context of working from home during the COVID-19 pandemic is scarce, thus it is interesting to further investigate. However, we were able to find the study by Russo, Hanel, Altnickel, and van Berkel who investigated the well-being and productivity of software professionals during the COVID-19 pandemic, and found that there was a lack of autonomy and competency development during the pandemic, which negatively affected the software professionals’ motivation and well-being (Russo, Hanel, Altnickel & van Berkel, 2020, p. 8). Another study by Butler and Jaffe (2020) also touched upon motivation in ISD during the pandemic, as they investigated how software engineers experienced working from home during the COVID-19 pandemic, and found that one of the challenges they experienced was lack of motivation (Butler & Jaffe, 2020, p. 9). It is important to mention that there is still a lack of research that explicitly investigates motivation in ISD, as the previous research only investigates it in continuation of other subjects such as well-being.

2.3 Literature on Control and Motivation

Studies both within and outside ISD have already established a series of important links between control and motivation. However, the majority of those studies have only researched the causal relationship between the two variables, especially in terms of control influencing the motivation of employees. A study that researches this is by van der Kolk, van Veen-Dirks, and ter Bogt, who found that controls can be used to motivate employees and bring forth desired employee behavior. They argue that different control modes influence both the intrinsic and extrinsic motivation depending on the control type (van der Kolk, van Veen-Dirks & ter Bogt, 2019, p. 901). Intrinsic motivation refers to behavior that is driven solely by personal satisfaction and enjoyment, whereas extrinsic motivation refers to behavior that is done in order to gain external rewards (Legault, 2016, p. 32). Also, a study by Goldbach and Benlian investigated how self-control and clan control affected developers' intrinsic motivation and found that both self-control and clan control had positive effects on software developers' intrinsic motivation (Goldbach & Benlian, 2015, p. 11). Even though most studies suggest that the relationship between control and motivation is solely causal, Johnson, Muraven, Donaldson and Lin argued that motivation can also have an impact on the enacted control (Johnson, Muraven, Donaldson & Lin, 2018, p. 123). They found that an increased motivation led to employees to exert self-control (Johnson et al., 2018, p. 122). Thus, we argue that the relationship between control and motivation should not be described as causal, but as transactional (Elkjaer & Simpson, 2011, p. 69), since the two variables affect and influence each other. This transactional relationship has not been examined in an organizational context nor an ISD context, where there has been a focus on how control affects motivation.

Orlikowski found a connection between the ISD controls that managers implement and subsequent effects on employees' well-being. In situations where employees feel that an ISD activity or tool slows or hinders their actions, they may experience diminished motivation or seek ways to workaround the control to re-establish their autonomy, which undermines the effectiveness of the control and the resulting outcome (Orlikowski, 1991, p. 31). However, other studies suggest that some ISD control environment and control enactments can encourage intrinsic and extrinsic motivations that influence an individual's participation in tasks associated with ISD and increase the feeling of satisfaction, which can ultimately lead to improved outcomes (Roberts, Hann & Slaughter, 2006, p. 996; Santana & Robey, 1995, p. 30). In addition, Remus et al. investigated how different control modes and controls styles relate to controllees' tasks performance and job satisfaction (Remus et al., 2016, p. 1). Even though the study regards job satisfaction, existing research has shown interrelatedness between job satisfaction and motivation (Theresia, Lahuddin, Ranti & Bangun 2018, p. 9; Dartey-Baah & Harlley, 2010, p. 13), thus it has been included in this research background. Remus et al. found that the use of formal controls is positively related to the controllees' job satisfaction (Remus et al., 2016, p. 14). Contrary to their hypothesis, the data pointed to a positive effect of formal controls on job satisfaction, however, not significant. This may be due to formal control can make the controllee's project roles less ambiguous (Remus et al., 2016, p. 15). They argued that the controllee may respond well to the controller's use of formal controls, improving their job satisfaction. However, at a certain limit, the controllees can feel 'over-controlled' which can negatively affect their job satisfaction (Remus et al., 2016, p. 15).

The review of the literature on control in ISD shows there is a need for more studies which focus on the controllees' perspective and studies which investigate control in the context of a pandemic. In regards to the review of the literature on motivation in ISD, we found that the motivating factors vary between studies depending on the context in the software organization. The context can refer to an internal context like the use of development practices and management styles, as well as external contexts, like the time period or global events. As there has not yet been conducted studies on software developers' motivation within the context of the COVID-19 pandemic, it is a research gap to be studied. As presented in the prior section, the relationship between control and motivation is transactional, thus when analyzing both control and motivation of the software developers, we will interpret how these can relate and influence each other. The transactional relationship had not been researched in an organizational context nor ISD context, which means there is a research gap. These research gaps found in this research background will be addressed in this study. This will be done by researching how managers can motivate and control software developers while working from home during the COVID-19 pandemic.

3. Theories

In this chapter, we present the theoretical framework which will be used to analyze the empirical data. In order to answer the research question, Kirsch's control modes will be used to investigate control, while Herzberg's motivation-hygiene theory will be used to investigate motivation.

3.1 Control Theory

Control theory aims to explain how an individual or group in an organization can ensure that another individual or group works toward and attains organizational goals (Kirsch, 1996, p. 1). Following Remus et al., we define control as; "*any attempt to ensure that individuals working on a project act in a manner consistent with organizational goals*" (Remus et al., 2016, p. 3).

Within control theory, the group will be using control modes by Laurie J. Kirsch, who is a researcher of control enactment in the information systems context (University of Pittsburgh, n.d.). Control modes refer to the overall characteristics of the controls (Cram & Wiener, 2018, p. 715). In relation to the research question, we try to understand the control modes enacted and employees' motivation in the organization, while the employees work from home. This can potentially give us an insight into how control and motivation are related while the employees are working from home. The use of control modes will supplement with various definitions, properties, and examples that can help indicate what kind of control the organization uses in the empirical data. It should be noted that most organizations do not solely rely on just one control mode. Instead, an organization typically uses multiple control modes in various degrees. In addition, prior studies on control have used control modes to describe control enactments, as shown in the literature review in chapter 2. Each mode can be characterized as formal or informal control (Kirsch, 1997, p. 217).

Formal control

Formal control is viewed as a performance evaluation strategy and is enacted by levels of management (Quain, 2018). For instance, a department manager manages the project managers, whereas the project managers manage teams. The controller's use of formal control provides the controllee(s) with a required level of guidance and structure, which consequently can result in improved work performance of the controllees (Remus, Wiener, Saunders & Mähring, 2020, p. 138). There are two modes within formal control: *behavior* and *outcome* (Kirsch, 1997, p. 217).

Behavior control is defined by the use of rules and processes, which if followed, will lead to desired outcomes. Controllers observe the behaviors of the controllees, who are rewarded based on the degree to which they follow these desired procedures. In addition, behavior control is implemented when appropriate behaviors are known or when controllee behaviors are observable to the controller (Kirsch, 1997, p. 217). For instance, a project manager requires a team to employ a detailed systems development methodology, where each step of how to successfully develop a system is precisely articulated. The manager controls the employees by either rewarding or punishing individuals according to how faithfully they follow the methodology.

Outcome control describes the use of predefined goals between the controller and the controllee. The completion of these goals are measured and rewards or penalties are distributed based on how well the goals were reached regardless of the controllers' process or behavior used to reach the goal (Kirsch, 1997, p. 217). Outcome control can be exercised in many ways including performance targets, milestones for a particular activity, or specific project goals. An example of outcome control, is a project manager who requires software developers to complete user stories assigned to them each week.

Informal control

Informal modes of control are based on social or people strategies. Because formal controls are hierarchical, where the decisions are made high up in the chain of command, they often ignore interpersonal or self-regulating dynamics that also govern behavior (Kirsch, 1997, p. 217). In addition, informal controls are typically undocumented and differ in levels of how many people it affects (Kirsch, 1997, p. 217). Kirsch presents two modes that represent different levels: *clan* (group level) and *self* (individual level) controls.

Clan control is implemented by promoting a set of common values, beliefs, and philosophies within a clan, which is defined as a group of individuals who are dependent on one another and who share a set of common goals (Kirsch, 1997, p. 217). By carefully selecting and socializing members, the clan creates norms and values, resulting in a group of individuals who share a common ideology, who have internalized a set of values, and who are committed to the clan. Instead of requiring employees to follow a written set of procedures, the socialization process, as well as rituals and ceremonies, serve to identify and reinforce acceptable behaviors. Thus, each group member can effectively function as both the controller and controllee. Further, specific goals and objectives are typically unknown at the beginning of an activity, but instead evolve over time. This is because clan control is often implemented when behaviours and outcomes are ambiguous and difficult to measure in an organization (Kirsch, 1997, p. 218). An example could be a project manager facilitating online team building events to encourage collaboration in a software development team that has started to work from home due to the pandemic.

Self-control is the second mode of informal control, which is where an individual sets their own goals, monitors their own work, and rewards or sanctions themselves accordingly (Kirsch, 1996, p. 3). Individuals who enact self-control are intrinsically motivated to achieve their objectives, define their own goals and processes appropriate for the task, and obtain feedback to self-evaluate their own performance (Kirsch, 1996, p. 3). With clan control, monitoring, rewarding and sanctioning are done across the group, whereas with self-control, it is up to the individual. Self-control is good for tasks that involve a great deal of autonomy, creativity, or problem solving, since it can be difficult for controllers to identify appropriate behaviors for them (Kirsch, 1996, p. 3). To give an example of self-control; a programmer wants to get better at documenting his codes, because he believes that it is critical to have extensive documentation. While documenting the codes, he also monitors how well he is progressing over time and intrinsically rewards himself for completing the job.

3.2 Motivation Theory

Motivation is an important factor of personal and organizational accomplishments, which has led to numerous studies and theories about motivation in the workplace (Kanfer et al., 2008, p. 4). Following Kanfer et al. , we define motivation as; *"a psychological process that influences how personal effort and resources are allocated to actions pertaining to work, including the direction, intensity, and persistence of these actions"* (Kanfer et al., 2008, p. 5). The motivation theory, which will be used in this study, is Herzberg's motivation-hygiene theory.

The group chose to use Herzberg's motivation-hygiene theory to examine the motivation of software developers, who were working from home during the pandemic. The reasoning behind the choice of the theory is because it explains that certain factors can lead to motivation. These factors can be implemented through control enactments, which can then lead to motivated employees. Even though the theory was from 1959, and thus might be outdated, it is still relevant for the global software development context (Beecham, 2014, p. 252). Furthermore, the theory has been applied in studies within several fields for decades, which suggests that it is a well-established theory on motivation.

Frederick Herzberg developed the motivation-hygiene theory as a response to the then-current perception of satisfaction being a single continuum, where satisfaction is represented at one end, while dissatisfaction is represented at the other end. Herzberg's motivation-hygiene theory had another approach to job satisfaction with two different continuums. One continuum regarded the hygiene factors which had dissatisfaction at one end and no-dissatisfaction at the other end, while the second continuum regarded the motivation factors that had no-satisfaction at one end and satisfaction at the other end. The two factors will be elaborated in the following subsections.

Hygiene factors

The hygiene factors are factors extrinsic to the work itself and have the purpose of preventing and reducing dissatisfaction among employees. These extrinsic factors help differentiate between dissatisfaction and no-dissatisfaction (Herzberg, 1993, p. xiii) and do not provide satisfaction or lead to increased motivation (Herzberg, 1993, p. 113). Managers can remove employees' dissatisfaction by providing sufficient hygiene factors that meet the employees' basic needs and then use motivation factors to provide satisfaction to meet the employees' higher-level needs (Ozguner & Ozguner, 2014, p. 213). The hygiene factors include *job supervision, interpersonal relations, physical working conditions, salary, company policies and administrative practices*, and *job security* (Herzberg, 1993, p. 113). Salary and job security did not appear in the empirical data, thus will not be elaborated nor included in this study.

Job supervision refers to the competence or incompetence, and fairness or unfairness of the managers or superiors (Herzberg, 1993, p. 47). A competent or fair manager can minimize dissatisfaction among employees, while an incompentent or unfair manager can lead to dissatisfaction. An example of job supervision that minimizes dissatisfaction could be a project manager allowing a software developer having the day off after working over time. This will lead to no dissatisfaction, as it will meet the software developer's basic need of having a fair manager.

Physical working conditions refers to the tools and environmental surroundings of where the employees work (Herzberg, 1993, p. 48). The physical working conditions can either be dissatisfying or not dissatisfying depending on the tools and surroundings being good or bad. Implementing good surroundings or having effective tools can eliminate or minimize dissatisfaction, which will then give the opportunity of fostering motivation among employees. For instance, a software developer having multiple screens while working from home might eliminate dissatisfaction, as multiple screens are needed in order to work efficiently. This fulfills a basic need of having good physical working conditions.

Interpersonal relations refers to employees' relations and interactions with superiors, subordinates, and peers at the workplace (Herzberg, 1993, p. 46). An example of how to promote interpersonal relations could be facilitating teamwork to create better relations between employees. For instance, a software developer being told by their manager to elicit requirements in collaboration with their team. According to the theory, this would minimize the software developer's dissatisfaction, as it would meet the software developer's basic need for relations with colleagues.

Company policies and administrative practices refers to effective or ineffective organization of work, beneficial or harmful personnel policies, agreement or disagreement with company goals, and high or low company status (Herzberg, 1993, p. 145). Good company policies and administrative practices can be used to eliminate or minimize dissatisfaction among employees. An example could be a clear guideline about how software developers should work from home, when they are not in the office. This will eliminate uncertainty among the software developers in the organization, and thus minimize the dissatisfaction.

Motivation factors

The motivation factors are factors intrinsic to the work itself and lead to positive job attitudes, as they satisfy the employees' self-actualization needs (Herzberg, 1987, p. 114). These intrinsic factors differentiate between no-satisfaction and satisfaction and can lead to increased motivation among employees. In order to do that, managers need to sufficiently fulfill the employees' basic needs to then further apply motivation factors to satisfy and motivate the employees (Ozguner & Ozguner, 2014, p. 213). The motivation factors include *achievement, recognition, work itself, advancement, growth opportunities,* and *responsibility* (Herzberg, 1987, p. 9). Achievement and advancement will not be included in this study, as they did not appear in the empirical data.

Recognition refers to the act of recognition given to employees by superiors, subordinates, or peers (Herzberg, 1993, pp. 44-45). This recognition can be positive, where employees get praised or noticed at work, or it can be negative where employees get blamed or criticized at work. The positive recognition will lead to satisfaction and foster motivation, whereas the negative recognition will not satisfy nor motivate employees. An example of positive recognition could be a manager praising a software developer for implementing virtual pair programming, while the whole team is working from home. This will satisfy, and thus motivate the software developer, as it is a recognition of their work.

Work itself refers to the actual doing of the job or work tasks in which the employees can either have good or bad feelings about (Herzberg, 1993, p. 48). According to the theory, employees having work tasks that evoke good feelings will lead to satisfaction, and thus increased motivation. An example of this could be software developers getting interesting or challenging tasks while working from home.. This will possibly lead to the software developers being more motivated, as they will feel good about their work.

Growth opportunities refers to the opportunity of an employee's growth in skills and status within an organization (Herzberg, 1993, p. 44). This can lead to satisfaction and motivation among employees. An example could be a manager telling a junior software developer to work on a difficult task in order to become more skilled at their job. The opportunity for skill development will lead to satisfaction and increased motivation, as the developer can become more skilled by performing a difficult task.

Responsibility refers to the responsibility and authority given to an employee by the company or a superior (Herzberg, 1993, p. 47). According to the theory, giving responsibility and authority to employees will lead to satisfaction and increased motivation. For instance, a manager telling a software developer in a team to be responsible for their own work. This would make the software developer feel trusted and also feel a sense of responsibility, which would lead to satisfaction and increased motivation.

4. Methodology

In this chapter, there will be a presentation of the methods applied in this study. This includes the case study approach, collection of the data, analysis approach, and lastly, how we establish the codes' reliability for the analysis.

4.1 Case study

In this study, the case study approach was applied to investigate how managers can motivate and control software developers who work from home. According to Lazar, Feng, and Hochheiser, a case study is; *"a detailed examination of one or more specific situations"* (Lazar, Feng & Hochheiser, 2017, p. 156). The approach focuses on examining phenomena in an applicable context that are beyond the researchers' control. This means that the researchers can develop a detailed understanding of phenomena, as they occur in the real world (Lazar et al., 2017, p. 157). For instance, in our case, we cannot answer our research question without examining a phenomenon within its real-life context, since the results from a simulated and controlled setting will not reflect the real world.

Within the case study approach, an important aspect is to use qualitative data that can help describe or explain behavior (Lazar et al., 2017, p. 157). In this study, we used a qualitative approach to explain how managers motivate and control software developers who are working from home. To answer this question, it requires data that can shed light on the enacted control by management and the motivation of the software developers which cannot be answered by solely including quantitative components, such as measuring how many meetings software developers have with their manager. This will not provide a sufficient understanding of the context, thus we need to obtain the data by having in-depth discussions about the context.

According to Yin (2014), case studies fall into one of three categories: *exploratory*, *explanatory*, and *description,* which each is used for different purposes (Lazar et al., 2017, pp. 159-160). As presented earlier, this study's research question is formed as a how-question, which is why the group decided on using the explanatory case study approach. Explanatory case studies focus on explaining and describing certain phenomena, but can also be used to explain causal relationships and to develop a new theory (Mills, Durepos & Wiebe, 2010, p. 370). In this study, the explanatory case study approach will be used to explain the transactional relationship between motivation and control enactments and not the causal relationship, since the two variables are seen as influencing each other. The approach will also be used to generate knowledge about motivating and controlling software developers that are working from home. Furthermore, we used a single case rather than multiple cases as the goal of this study is not to generate generalized results, but a focused investigation of one specific organization. This was because organizations are distinctive and have different organizational structures and cultures that could have an influence on the employees' motivation and management's control enactments, and thus make a comparison to other cases difficult (Lazar et al., 2017, p. 169).

4.1.1 Case selection

According to Flyvbjerg, the form of sampling and strategy for case selection depends on the purpose of the study (Flyvbjerg, 2006, p. 229). For example, when the objective of the study is to achieve a representative sample that can be generalized for the entire case population, a random sample or representative case should be used (Flyvbjerg, 2006, p. 230). However, in our study, we want to investigate a specific phenomenon through the use of a single case study. Therefore, the strategy used to select the case for this study was the *information oriented selection*. This means that the case was selected on the basis of expectations about their information content, which is in contrast to random sampling. By using this strategy, we maximize the utility of information from small samples and few cases. Since the research question does not seek to generalize outside of the specific case, the information oriented selection was deemed most useful as we want to obtain the maximum amount of information on the given phenomenon (Flyvbjerg, 2006, p. 230).

4.1.2 Presentation of WindCo

As mentioned earlier, the following case was selected with the purpose of investigating how managers can motivate and control software developers who work from home during the pandemic. In order to make sure that no confidential information from the company was shared, the group decided on an alias as a substitute for the name of the company. Thus, throughout the project, the organization is referred to as *WindCo*. WindCo is a Danish wind turbine manufacturer that designs, manufactures, installs, and services wind turbines across the world. The organization is internationally recognized and has departments in several countries such as Portugal, China, and India. The employees at WindCo are familiar with remote working, as they have to collaborate with each other across these countries. Additionally, the employees are used to working from home, since the organization encourages flexibility. As a result, it was acceptable among the employees to work from home a couple of days a month prior to the pandemic. The studied team used Scaled Agile Framework (SAFe) during the pandemic, where they worked with different teams in both their own department and other departments in an

Agile Release Train (ART). In the ART, they have Program Increment (PI) Planning which is held every second month and lasts two hours. At this meeting, the different teams present and evaluate their projects, and a plan for the next PI Planning is presented. After a PI Planning session, the teams hold a Sprint Planning session, where they plan the Sprint which they then execute for the next two weeks.

The studied subjects from WindCo were a Submodule Owner, who acted as a department manager, and the software developers; Data Scientist, Lead Financial Engineer, and a Data Analyst, who all analyzed the data and signals from the wind turbines while they were working from home. It was decided to include the Submodule Owner in our research, as we were aware that the managerial intentions and subordinate perceptions can differ even when talking about the same control enactment. Therefore, we have complemented the three interviews with the software developers with one interview with the department manager. In this way, we were able to obtain a more balanced understanding of the control enactments, although our main focus remained on the software developers' perceptions on the control enactments. In this study, the Submodule Owner was viewed as the controller, whereas the software developers were the controllees. It should be noted that both the Scrum Master and Product Owner were not included in this study as studied participants, but were still considered as controllers, as they showed in the empirical data. As mentioned earlier in section 3.1, the controller's use of formal control provides the controllee(s) with the required level of guidance and structure, which consequently can result in improved work performance of the controllees. Since the Scrum Master ensured and therefore also guided the team members in complying with the Scrum rules and additionally facilitated the Scrum events, the Scrum Master was viewed as a controller in this study. In addition, since the Product Owner prioritized the tasks most important to the customers, he was able to guide the software developers in the right direction of the product, thus he was also considered to be a controller in this study.

4.2 Empirical Data and Data Collection

Qualitative interviews were conducted to collect the empirical data, as a deep understanding of the phenomenon was needed in order to answer the research question. With qualitative interviews, we can get an insight into the interviewees' experiences, thoughts, and feelings (Patton, 2015, p. 426). Within qualitative interviews, a semi-structured design was chosen, since it gives the researchers the option of utilizing an interview guide with pre-made questions. Moreover, the researcher also has the opportunity to change the sequence and forms of the questions, which makes it possible to explore new aspects based on the interviewees' answers (Kvale & Brinkmann, 2009, p. 124). Within qualitative research a core concern is saturation, which is used to determine whether the data is adequate enough to have gained a robust and valid understanding of the studied phenomenon (Hennink & Kaiser, 2019, p. 1483). Therefore, the main goal of the interviews was to achieve data saturation, which can be reached when the ability to obtain additional new information is not feasible (Fusch & Ness, 2015, p. 1413).

4.2.1 Interview Guide

The interview guides used for the semi-structured interviews, which can be found in Appendix 1, consisted of dynamic and thematic questions. The aspect of theme refers to the questions' relevance for the research theme, whilst dynamic regards the interpersonal relationship in the interview (Kvale & Brinkmann, 1996, p. 129). It was important that they were dynamic as the group wanted a natural interaction with the interviewee. Therefore, the questions were formed to be easily understood to avoid overly academic formulations. Additionally, the questions were thematic since the majority of the questions were based on control and motivation theory, which is in accordance with the theme in this study; how software developers are motivated and controlled. We designed two interview guides, where the questions and sequence were partially fixed. The decision on making two interview guides was due to the differences between a controller and controllee, which required a set of questions to each role in order to sufficiently explore the phenomenon. According to Patton, an interview guide is used to make sure that the same questions are pursued with each person interviewed (Patton, 2015, p. 439). By using an interview guide, the data collection becomes more comprehensive since the interview guide can serve as a checklist to make sure that the relevant themes are covered during the interview (Patton, 2015, pp. 438-439). Furthermore, the data collection becomes more systematic, as the themes and issues to be covered are specified beforehand (Patton, 2015, p. 439).

4.2.2 Remote Interview

The interviews were conducted remotely due to safety concerns, as the government discouraged physical interaction. Interviewing remotely meant that the group and interviewees were not co-located during the interviews, thus communicating through teleconferencing software (Trate et al., 2020, p. 2421). The interviews were conducted through Microsoft Teams, which also made it possible to record them. By recording the interviews, we were able to capture the interviewees' own actual words rather than relying on notes to convey the overall meaning (Patton, 2015, p. 472). When conducting interviews online, the nonverbal communication is lost (Patton, 2015, p. 494). In order to accommodate this challenge, we conducted the interviews using the webcam function on Microsoft Teams. This enabled us to notice the nonverbal cues, which improved the overall communication between us and the interviewees.

4.2.3 Transcription

The group chose to transcribe the recorded interviews since transcription is vital for qualitative research. As previously mentioned, we gathered important data through the recorded interviews, however, the data needed to be sorted out and organized before we could make sense of it. In this instance, transcribing the qualitative data was important because it created a text-based version of the recorded interviews. This made it possible for the data to be arranged more systematically, thus easier to analyze and share as well as preserving the accuracy and integrity of the data. We decided on specific guidelines to follow prior to the transcription, since all members of the group were going to transcribe the recorded interviews. These guidelines helped us ensure that the transcripts were similar in structure and content, which would make it easier to navigate through, when we had to analyze the data later on. As part of our guidelines, we chose to transcribe in the style of intelligent verbatim, which is where fillers, pauses, 'um' and 'er', repeated words, and non-verbal communication are excluded (Chege, 2009). This style was chosen since the group wanted to focus on what the interviewees said rather than the visual data. This corresponds well with the intelligent verbatim style as the main purpose is to promote more readable transcripts, whilst still staying true to the content of the interviewees (Chege, 2009).

4.3 Qualitative Content Analysis

In order to analyze the empirical data, the group chose to conduct a qualitative content analysis to uncover control modes, hygiene factors, and motivation factors in the WindCo case study. Qualitative content analysis is a research method, where subjective interpretation of the text data's content is done by coding and identifying themes or patterns (Hsieh & Shannon, 2005, p. 1278). According to Downe-Wamboldt, the goal of the content analysis; *"is to provide knowledge and understanding of the phenomenon under study"* (Downe-Wamboldt, 1992, p. 314). In this study, we want to understand how managers can motivate and control software developers who are working from home during the pandemic by analyzing transcripts of the conducted interviews.

There are three approaches to qualitative content analysis: *conventional*, *directed*, and *summative* (Hsieh & Shannon, 2005, p. 1278). We decided on the directed approach since it is guided by a more structured process, which is useful for analyzing multiple transcripts. Moreover, the approach uses existing theory or prior research to provide predictions about the variables of interest or the relationships among variables which can help determine the initial coding scheme (Hsieh & Shannon, 2005, p. 1281). In our case, the variables of interest are the passages in the transcripts that concern motivation and control, which will be identified with the help of an initial coding scheme based on control modes and Herzberg's motivation-hygiene theory. In addition, we will also include passages that cannot be categorized in the initial coding scheme and give them new codes. This final coding scheme can be found in Appendix 3, where we presented the codes with descriptions and examples to provide a shared understanding of the variables.

In this study, we chose the coding strategy by Hsieh and Shannon (2005), where the researchers first identify all the instances of a particular phenomenon by highlighting relevant passages without coding on the first impression. In the next step, the researchers code all the highlighted passages using the predetermined codes, whereas the passages that could not be categorized with the initial coding scheme will be given new codes (Hsieh & Shannon, 2005, p. 1281). This was done as we wanted to be able to come up with new codes and not be dependent on the initial coding scheme. By using this strategy, it will increase the trustworthiness of the analysis (Hsieh & Shannon, 2005, p. 1282), as we will not be biased and solely dependent on the predetermined codes.

4.3.1 Negotiated Agreement

In order to establish the reliability of the codes, we chose to implement the *negotiated agreement* approach to the coding process. Reliability is an important part of content analysis, since a goal of content analysis is to identify and record relatively objective characteristics of the data. Thus, if the reliability is not established, the interpretations of the data cannot be considered valid (Lombard, Snyder-Duch & Bracken, 2002, p. 589). In this study, we focus on the type of reliability that is reproducible across coders, which is also known as *intercoder reliability*. Within intercoder reliability, the main concern is to examine whether different coders would code the same data in the same way (Campbell, Osserman, Quincy & Pedersen, 2013, p. 295). Therefore, the negotiated agreement was deemed appropriate to use, since it assesses the intercoder reliability by having two researchers separately code one transcript. Subsequently, the two researchers would compare the codes, and lastly discuss their disagreements in an effort to reconcile them, thus avoiding as many discrepancies as possible (Campbell et al., 2013, p. 305). In our case, we had four coders coding each transcript. Thereafter, negotiating about the codes in order to arrive at final coded transcripts. These coded transcripts can be found in Appendix 2. To give an example of a negotiated code, the Data Scientist mentioned:

"I am not tracking any personal progress, as such other than what is part of the whole agile framework where we track how many so-called stories are completed within each feature, and how many features have we completed and such. So, you can say, we track our progress as a team."

- Data Scientist (Appendix 2.2, 26:15-28:05).

This instance was coded as 'Clan Control' by two coders, whereas the two other coders coded it as 'Outcome Control'. Through negotiation, the coders collectively agreed that the code 'Outcome Control' was more suitable for the chosen passage. This was due to the fact that the Data Scientist explicitly talked about the agile framework, which has controls for monitoring the output of the controllee. Since the focus is on how the team is collectively monitored through these agile practices, it made sense to code it as 'Outcome Control' rather than 'Clan Control'.

5. Data Analysis and Findings

In this chapter, we will analyze the case by identifying control modes, hygiene and motivation factors in relation to WindCo's agile work process and performance management.
As WindCo made use of the agile method, in which we could find all Kirsch's control modes, it was important to see how these control enactments could be related to the software developers' motivation or vice versa. We will look at how these control enactments had an effect on the software developers' motivation while they worked from home, but also how their motivation may have had an impact on the control enactments. This makes it possible to answer our research question, in that our results of the analysis will show which types of control motivate the software developers while they worked from home, and which controls did not. In that way, we canprovide suggestions for what WindCo can do in order to motivate and control the software developers while they work from home during the pandemic.

5.1 Agile Work Process

The work process is important to investigate when researching how software developers can be motivated and controlled while working from home. A work process can be defined as a series of work activities completed across individuals and functional areas of organizations to achieve organizational goals (Jacobs, 2019, p. 18). Often during these work activities, there are control enactments to ensure that employees are behaving according to organizational frameworks, which could influence the employees' motivation. As mentioned in section 4.1.2, WindCo used an agile framework when the software developers were working from home. Thus, there will be a focus on the agile work process in this theme.

5.1.1 PI Planning sessions

Within WindCo's agile work process, SAFe was used, where they had PI Planning sessions. These sessions can be categorized as behavior control, as they are usually facilitated by a Release Train Engineer (RTE) (Scaled Agile, 2021a), who serves as a coach for the ART (Scaled Agile, 2021b). The RTE's responsibilities are to facilitate ART events and processes, and also assist the teams in delivering value (Scaled Agile, 2021b), thus the RTE is able to observe the activities of the teams within the ART during the PI Planning sessions. At these sessions, which took two days, teams from different submodules came together and presented and evaluated what they had done in the previous PI Planning (Appendix 2.4, 01:22-06:17). They had demos, where they were able to see what the teams had been doing, and how they were progressing (Appendix 2.1, 39:36-41:29). Also, a plan for the next PI Planning would be presented, where the Product Owners, who also act as team leaders, also present the priorities (Appendix 2.1, 02:47-05:38; Appendix 2.2, 04:25-05:29), and the different teams would split into their own meetings to estimate and divide the tasks in sprints. Here, they would also be able to find the dependencies (Appendix 2.4, 01:22-06:17). In the following quote, it can be seen that the Data Scientist had a positive attitude towards PI Planning sessions.

"I guess we do it by talking a lot about the tasks and coordinating and also our team leader, when we do these planning stations, our team leader spends some time on presenting the upcoming features that we have been assigned to. Presenting what are the thoughts behind this and what are the expectations from external stakeholders, and where do we want to go in general and how does it fit into the bigger picture. So, that helps a lot in seeing the purpose of the things we do."

- Data Scientist (Appendix 2.2, 12:30-14:55)

The Data Scientist mentioned that it helped a lot in seeing the purpose of their work, when their Product Owner presented; the upcoming features, the thoughts behind these, the expectations from external stakeholders, where they should go, and how that fits into the bigger picture, during the PI Planning sessions. Also, since all the other teams in the ART presented their progression of their projects at the PI Planning sessions, the software developers were able to get an overview of all the projects in the ART, which helped them to get a better understanding of what every team had worked on, and how their own project fitted into the whole process. All of this helped the software developers to better understand the purpose of their work while they worked from home. Getting an overview of the ART's progress from home would otherwise have been difficult to obtain without online PI Planning sessions, since the software developers were working alone from their homes, and there were no other activities that facilitated this. Furthermore, since all teams from the ART attended the PI Planning sessions, the software developers were also able to connect with the other colleagues from the other teams besides their own team members, in which they already talked to everyday. This could make the software developers feel like they were a part of a greater community, which can be difficult when they always worked alone from home. The software developers were for these reasons not dissatisfied in regard to the company policies and administrative practices, in which PI Planning sessions came from. According to Herzberg, agreeing with a company's policies and administrative practices does not lead to motivation, but it fulfills one of the basic needs, which can be sufficient in order to be able to foster motivation. Even though the PI Planning sessions had positive effects on the software developers, it can also be seen that they were difficult to carry through from home. This can be seen in the following quote by the Data Analyst.

"If there are meetings that are taking the whole day, then it is better to split it up. For example, the PI Planning goes for two days. So, for two full days we are sitting online in front of the computer, two full days of meetings. Yeah. I think maybe that could change that."

- Data Analyst (Appendix 2.4, 38:45-39:16)

The Data Analyst was not fond of the PI Planning session stretching for two days, when they worked from home and thought it would be better if the session could be split up. When having a PI Planning session from home, they had to sit online in front of the computer for two full workdays. This was especially draining, as it was two full days of meetings, meaning that they had to sit at their desk and follow along multiple meetings on their computer. In addition, virtual meetings removed the active interaction, which made it more difficult for the software developers to keep their concentration and be engaged in the conversations or discussions. Therefore, having long meetings, such as PI Planning sessions, from home was not optimal for the software developers, as the physical working conditions were poor. According to Herzberg, this led to the software developers feeling dissatisfied in regard to the physical working conditions, which can hinder the fostering of motivation.

It can be seen that the PI Planning sessions helped the software developers to better understand the purpose of their work while they worked from home, as the Product Owner presented thoughts behind upcoming features and so on and the software developers got an overview of the ART's progress. Furthermore, they felt that they were a part of a greater community, as they were able to connect with other colleagues besides the ones in their own team. For these reasons, they were not dissatisfied with the company policies and administrative practices, which PI Planning sessions came from. However, even though PI Planning sessions were helpful, they were draining to have online, as the software developers had to sit at their desks and have online meetings for two full workdays. As they had to have long meetings such as PI Planning sessions from home, it made their physical working conditions dissatisfying.

5.1.2 Sprint Planning

After the PI Planning sessions, the software developers in the Scrum team would conduct Sprint Planning, which was part of the agile method. The Scrum team had Sprint Planning on the first day of each Sprint, with three to four Sprints per Program Increment (PI). After the PI Planning, the Scrum team had evaluated the effort it would take to finish the tasks, broken down large tasks into smaller tasks, prioritized them based on importance, and divided the tasks into the upcoming Sprints (Appendix 2.3, 13:51-14:02; Appendix 2.4, 01:22-06:17). This was followed up by the first Sprint Planning, where the software developers would delegate tasks amongst themselves. (Appendix 2.4, 01:22-06:17). In this study, the Sprint Planning is categorized as clan control, even though it would normally be considered a formal control, as both the Product Owner and Scrum Master would be present during this activity (Schwaber & Sutherland, 2020, p. 8). However, Sprint Planning was perceived as a clan control by the software developers, since the software developers were functioning as both controllers and controllees promoting a common value within the team, which was the shared responsibility. This was evident when the software developers were delegating tasks during the Sprint Planning, as seen in the following quote.

> *"So, we are sort of given a frame, and then we have the freedom and responsibility to make a solution within that frame and within those requirements. We almost have complete freedom, unless it is something that another depends on, and we, of course, coordinate with the others."*
>
> - Data Scientist (Appendix 2.2, 21:55-23:36)

The Data Scientist mentioned that the software developers were given a frame and requirements, which in this case were the prioritized tasks presented by the Product Owner during the PI Planning sessions. The software developers, as a team, then had the freedom and responsibility to break down these Sprint tasks, arrange, and delegate them among themselves. However, sometimes the tasks were dependent on other teams in ART, which meant that some tasks had to be prioritized, and the software developers in the Scrum team had to coordinate with the other teams. Even though there were given requirements and dependencies, the software developers were still able to decide how to break the Sprint tasks down, arrange, and delegate them among themselves, which gave them a sense of shared responsibility. Thus, the software developers were given responsibility in Sprint Planning during the pandemic, which motivated them according to Herzberg. Furthermore, the shared responsibility along with the delegation of tasks also made it possible for the software developers to choose tasks that were different from the tasks in the previous Sprints, as seen in the following quote.

"(...) But also just the different tasks that we have, motivated me. Some of it was something new, just getting some new experiences and knowledge, that just motivates me."

- Lead Financial Engineer (Appendix 2.3, 23:53-24:26)

The Lead Financial Engineer mentioned that they had different tasks while they worked from home, and that these motivated him, as they were new, which would let him get new experiences and knowledge. As the software developers were collectively responsible for delegating the tasks, the Lead Financial Engineer was able to choose tasks that he found interesting and made him feel good, which were tasks that were different or new. The software developers had the freedom to choose tasks that they found interesting, which meant that they were satisfied with the work itself. According to Herzberg, work tasks that evoke good feelings will lead to motivation. Thus, the software developers were motivated during the lockdown, as they had the opportunity of working with tasks that gave them new experiences and skills, thus not bore or tire them out and evoked good feelings about their work. This was especially important during the pandemic, as the software developers spent long hours in front of the computer at home without any physical contact with colleagues that could otherwise have made a workday more interesting. Being able to at least work with interesting tasks can therefore improve their situation at home. Even though the Sprint Planning had an effect on the software developers' motivation, it can also be seen the other way around. The software developers had to have the desire to be responsible in order to successfully conduct the Sprint Planning, as they were the ones to break down the tasks, arrange, and delegate them.

It was shown that the software developers benefited from conducting Sprint Planning from home, as they were given responsibility by sharing the responsibility of the activities in the Sprint Planning. Thus, they were satisfied with the responsibility they were given, and thereby motivated during the pandemic. Furthermore, the shared responsibility for delegating tasks gave them the opportunity to choose interesting tasks that could give them new experiences and skills. This evoked good feelings about their work, which also motivated them during the pandemic. Transaction was evident regarding Sprint Planning, as the motivation also affected the Sprint Planning. The software developers could successfully conduct the Sprint Planning due to their desire to be responsible for breaking down the task, arranging, and delegating them as they saw fit. Thus, in order to successfully conduct the Sprint Planning, the software developers must have a desire for responsibility as well.

5.1.3 Sprint

After the Sprint Planning, the Sprint would begin where the software developers would spend the following two weeks working on their tasks. During the Sprint, the software developers would enact self-control, as they were given autonomy to make independent decisions for their tasks, thus were consequently being able to monitor their own progress. When a software developer was assigned a task, he would be responsible for coming up with a solution that fits within what was agreed upon by the Scrum team during the Sprint Planning. If one was in doubt, he would be able to ask for the others' opinions (Appendix 2.2, 23:59-24:48), and receive help. In addition, the software developers were also more or less able to manage their own time, while working from home (Appendix 2.3, 19:32-20:47). Since the individual software developer was allowed to form solutions to their own tasks, thus being allowed to experiment with the solutions themselves, it led them to feel a greater ownership of their tasks. This could have made working from home better, as their tasks would be more interesting besides already being allowed to get the tasks they wanted to work with. Since the individual software developer was responsible for their own tasks, it meant that they were satisfied with the given responsibility for the management of tasks, thus also motivated according to Herzberg. Besides being allowed to be responsible for their own tasks, they were also allowed to manage their own time while working from home. This was perceived positively by the Lead Financial Engineer, which can be seen in the following quote.

> "*But yeah, the last couple of weeks, they have been okay, they have actually been pretty nice. You can more or less manage your own time, if you need to take a 15 minute break to get some air or something. Now, I am working from home with my girlfriend. We can have lunch together and stuff like that. So, that is actually nice*."
> - Lead Financial Engineer (Appendix 2.3, 19:32-20:47)

The Lead Financial Engineer described that it was nice to be able to manage his own time, as he could take a break whenever needed and was also able to have lunch with his partner. This meant that by being able to schedule his own workday, it made it easier for him to work from home, as he was able to take a break whenever it was needed for him. In this way, he was able to work at his own pace, thus avoiding feeling overworked. Also for the Data Analyst, it can be seen that work got easier, when he was able to manage his own time. This can be seen in the following quote.

"I try to arrange my work hours a little bit, so that I start early, and during the day so that I work a little bit less, and in the evening, when they are sleeping, then I also get some work time. So that there is more time in the more quiet hours to work."

- Data Analyst (Appendix 2.4, 10:30-11:48)

The Data Analyst described that he could manage his own time, as he worked in the early morning and evening, but worked less during the day. This was to avoid working during the hours where the children would be awake and possibly interrupt him. Since the Data Analyst was able to manage his own time, he could decide to work in the hours where it was quiet, which made it easier for him to focus on his work, and thereby easier to work from home. Being able to work in a quiet environment is especially important for software developers, as their work can consist of highly complex tasks which require them to be focused and uninterrupted. The fact that the software developers were more or less allowed to manage their own time, meant that they could schedule their own workday in which they saw fit. Thus, they were given responsibility for their own process, which was satisfying and thereby motivating according to Herzberg. This was especially important to them, as they were the ones who best knew how their days should be managed, so that they fit with their new surroundings, thus were also able to make their physical working conditions less dissatisfying. In addition, as the software developers were responsible for their own tasks and work schedules, there was never any doubt or questions about how many hours they worked daily. They were allowed to work the way they found the most suitable to their work situation at home (Appendix 2.2, 56:03-58:11). Even though the Sprints had an effect on the software developers' motivation, it can also be seen that their motivation for responsibility had an effect on the Sprints as well. The software developers must have a desire to be responsible in order for their Sprints to be successful, as they had to be responsible for their own work and time during the pandemic. If the software developers were not responsible for their own work during the Sprints, it would complicate the work process as they would be too dependent on each other. This would mean that the software developers would need to be constantly guided online, which would be more time consuming. Additionally, it would require the software developers to work closely together, as they would be dependent on each other, which would make it harder for them to adjust their schedules according to their situation at home. Thus, the Sprints would not be fully successful without responsible software developers.

Allowing the individual software developers to self-manage by letting them form their own solutions to their tasks, meant that they felt greater ownership of their tasks, as their tasks would be more interesting to them. As they were given responsibility for their own tasks, it satisfied them during the pandemic, and thereby also motivated them. In addition, since the software developers were also able to manage their own time, it also gave them responsibility and made their physical working conditions better, as they then could schedule their time as they saw fit with their new surroundings. The software developers also must have a desire to be responsible in order for their Sprints to be successful, as they had to be responsible for their own work and time during the pandemic.

5.2 Performance Management

Performance management is important to investigate when researching how software developers can be motivated and controlled while working from home. This is because setting employee performance goals, monitoring employee performance, and providing ongoing feedback to employees can motivate software developers to perform in line with corporate goals (Farndale, 2016, p. 325), and thus have an influence on their motivation. In addition, we have included social activities as part of the performance management, as they can have an effect on the software developers' work environment and consequently their performance. This is because social activities can lead employees to have good relations with their peers, which then can lead to better performance (Patricia, 2015, p. 119). On the other hand, employees who do not have good relations with their peers can lead to poor performance (Patricia, 2015, p. 118).

5.2.1 Daily Standup Meetings

One way to manage the performance of the software developers at WindCo was to have daily standup meetings, which was a part of the agile method. These meetings, which lasted 15-20 minutes every morning, involved the whole Scrum team (Appendix 2.1, 31:16-32:18), where the software developers would give a status on ongoing and future tasks (Appendix 2.4, 01:22-06:17). During these meetings, the Scrum team used a Kanban board in Jira to get an overview of the tasks and monitor the tasks' progress, whether they were 'not started', 'in progress', 'delivered', 'verified', or 'done' (Appendix 2.3, 16:59-18:10). Both the Scrum Master and Product Owner could pull tasks back and forth depending on the progress of the different tasks (Appendix 2.3, 18:20-19:17). In addition, the daily standup meetings also enabled the software developers to ask for help if they had any problems performing their tasks (Appendix 2.4, 01:22-06:17). The daily standup meetings can be categorized as outcome control, since the Scrum Master and Product Owner were present at these meetings, thus were also able to monitor the amount of output by the software developers each day.

The daily standup meetings were mainly perceived positively, which can be seen in the following quote.

> *"But in general, I think the time is really well spent, so everybody knows where they are going and can reach for help from the other, or from outside from our team if necessary. (...) As a side effect, I have the feeling, and I think that is also the purpose that it is not as much me as a person, who is responsible for feeding tasks. It is us as a team that delivers solutions to problems. For me, it gives me a much more relaxed work-life balance to know that everything does not rely on me, rather on the team. Also, when things are a little hard during the lockdown, especially with the children who get ill and such, it means we as a team can work around tricky situations where somebody becomes ill, or some parts take longer than expected"*
>
> \- Data Scientist (Appendix 2.2, 05:36-07:16)

The Data Scientist believed that the time was spent well, and it was good to know what everyone was doing, what they were going to do, and every team member could ask for help. Furthermore, he elaborated that having these daily standup meetings gave him a feeling of not being responsible for the tasks alone, but it was the whole Scrum team who shared the responsibility, which gave him a much more relaxed work-life balance. When sharing the responsibility, they were able to work around tricky situations where someone's child becomes ill or tasks taking longer than expected. If the Scrum team did not have these daily standup meetings while working from home, then it would be difficult for the whole team to know what each software developer was doing, as they were isolated in their own homes and thereby not being able to easily check up on each other. Thus, the company policies and administrative practices regarding the daily standup meetings did not dissatisfy the software developers, since they could check up on each other every day while they were working from home. In addition, as the daily standup meetings gave them the feeling that everyone shared the responsibility of the tasks, it also made them feel responsible, which satisfied them and thereby also motivated them during the pandemic. The shared responsibility also meant that they helped each other when things at home became difficult. The Data Scientist had children which made working from home challenging to some degree, as the children required care when they became ill (Appendix 2.2, 05:36-07:16). Also, the Data Analyst had the same concern while working from home, as he had children that needed help with their online classes (Appendix 2.4, 10:30-11:48). These interruptions from children made the software developers to some extent dissatisfied with their physical working conditions at home. However, with the help from others, they were able to rearrange tasks among each other, and thereby make the physical working conditions at home less dissatisfying. The daily standup meetings also made the software developers feel less disconnected and isolated from their team members, while they were working from home. This can be seen in the following quote.

"Sometimes, I think we missed the team members. Of course, having a physical presence. But usually, we are seeing each other online, sometimes on the video or at least, we hear each other out. So, I think, of course, we miss some physical presence. Otherwise, it has not been hard to work from home."

- Data Analyst (Appendix 2.4, 20:15-20:52)

The Data Analyst mentioned that he missed having his colleagues physically nearby, but having these daily standup meetings made it possible for him to connect with his team members. He could hear and sometimes see his team members during these online meetings, which would lessen the feeling of being disconnected from colleagues. For that reason, the software developers were less dissatisfied with their physical working conditions regarding having no colleagues nearby, since they had these daily standup meetings that made it possible for them to have a connection with their team members. Although the daily standup meetings had an effect on the software developers' motivation, it can be seen that the software developers' motivation also had an effect on the daily standup meetings. This was because the software developers' shared responsibility for their work also encouraged them to help each other when needed. Thus, the software developers must have this shared responsibility to successfully conduct the daily standup meetings.

It was shown that the daily standup meetings were useful, as the software developers could check up on each other every day and thereby get an overview of the Scrum team's progress. This led them to not be dissatisfied with the company policies and administrative practices. In addition, the daily standup meetings gave them a feeling of shared responsibility, which meant that they felt responsible during the pandemic and thereby motivated. Since they shared responsibility, it also meant that they helped each other when things at home became difficult, which made the physical working conditions less dissatisfying. Additionally, the daily standup meetings connected the software developers, which made their physical working conditions less dissatisfying, as they were able to talk and sometimes see each other virtually every day. Lastly, we found that software developers must have this shared responsibility to successfully conduct the daily standup meetings, as the shared responsibility encouraged them to help each other, which they did at these meetings.

5.2.2 Check-in Meetings

The software developers also had online check-in meetings, which was an activity that the software developers agreed on facilitating. They arranged check-in meetings when somebody had been working a lot on some parts of the project and wanted to share some knowledge about it with the others (Appendix 2.2, 12:30-14:55). In regard to the control modes by Kirsch, these meetings can be categorized as clan control, as they were the software developers' own initiative to facilitate whenever it was deemed necessary. It can be seen that the check-in meetings had a positive impact on the Data Scientist in the following quote.

> *"We arrange check-in meetings, where everyone has a chance to learn what has been done by the others in the team. So that everyone has some knowledge of what we are doing as a team (...) But it also means that we all have some sense of where we are going with our tasks and I guess that helps on motivation"*
>
> - Data Scientist (Appendix 2.2, 12:30-14:55)

The Data Scientist mentioned that through the check-in meetings, they had the chance to learn what has been done by the other software developers in the team. This meant that they were able to get new knowledge related to the other software developers' tasks and were also able to get a shared sense of where all the software developers were heading with their tasks. By conducting check-in meetings from home, the software developers were able to continue learning from each other, which resulted in their continuous growth in skills. As this was the case, it meant that they were satisfied with the growth opportunities during the pandemic, which according to Herzberg, motivated the software developers. In addition, the fact that the check-in meetings were facilitated by the software developers themselves, meant that the software developers had a need to share knowledge and learn from each other. This learning was important for the software developers, as it enabled them to easily step in and help each other during the daily standup meetings. By sharing knowledge within the Scrum team, each software developer would have a better understanding and overview of the tasks. This made it easier to reassign the tasks between them, so that the ones who were capable could step in for those who were not. It can be seen that the check-in meetings had a positive effect on the software developers' motivation. However, it can also be viewed the other way as their eagerness for growth in skills and knowledge could have resulted in them wanting to facilitate such meetings where they were able to learn from each other. This meant that the software developers needed to have an eagerness for growth in skills and knowledge for this control to be successful.

By having the check-in meetings, the software developers were able to continuously grow their skills while working from home, as they got new knowledge from sharing knowledge with each other. This motivated the software developers, as they were then satisfied with their growth opportunities during the pandemic. In addition, sharing knowledge also made it easier for the software developers to step in for each other when needed while working from home. Transaction was evident in this control, as in order for check-in meetings to be successful, the software developers must have an eagerness for growth in skills and knowledge.

5.2.3 One-to-ones

Another control that was a part of the performance management was the one-to-one meetings. These meetings were arranged by the Submodule Owner every second week, where he would meet with the individual software developer to talk about how things were going at work for the specific software developer (Appendix 2.2, 04:25-05:29). At these meetings, which lasted half an hour, they could talk about anything. Sometimes non-work related subjects, other times, serious conversations about career, tasks, and what was good and bad. In some instances, the Submodule Owner also used these meetings when he had something specific that he needed to discuss with the software developers about, but in general, it was very much up to the developers to set the agenda for the one-to-ones. The Submodule Owner tried to keep the one-to-ones a very open space, where the software developers had the opportunity to express their thoughts and be listened to (Appendix 2.1, 21:29-22:32). The one-to-one meetings can be categorized as clan control, as it was an initiative from the Submodule Owner to ensure that his subordinates were doing well at work. Thus, the one-to-one meetings were not a formal control, as they were not WindCo's own practice. In the following quote by the Data Scientist, it can be seen that the one-to-ones had a positive impact.

"*(…) I have my department manager, who I talk to every two weeks. He is really good at asking me, "how is it going with work?" and "what do you think of the tasks or overall directions?". I think that he is good at keeping the spirits high and keeping me and everybody else motivated. I think he is a really good leader and very invested in his employees.*"

- Data Scientist (Appendix 2.2, 15:11-18:31)

According to the Data Scientist, the Submodule Owner was good at keeping the spirits high and keeping everyone motivated by asking the software developers about their thoughts and feelings regarding their work.

Furthermore, he described that the Submodule Owner was a very good leader, who was very invested in his subordinates. Also, the Lead Financial Engineer acknowledged that the Submodule Owner was a good leader, as he was good at setting up small meetings just to check in (Appendix 2.3, 27:30-28:12). Since everyone was working from home, the Submodule Owner was not able to easily check on the software developers' well-being physically, however, he was still able to ensure the software developers' well-being during the pandemic. This was because he regularly arranged online one-to-ones, which gave the software developers the opportunity to also regularly share their worries or problems with the Submodule Owner, who then was able to take action if necessary. In addition, by having these meetings, where the software developers' opinions toward their work were in focus, the software developers felt that their opinions were recognized by the Submodule Owner. According to Herzberg, this recognition meant that they were satisfied and motivated. In addition, since the Submodule Owner was able to ensure the software developers' well-being during the pandemic, it led to no dissatisfaction of the hygiene factor, job supervision, which made it possible to foster motivation through motivation factors. The Submodule Owner showed that he was a competent manager, as he regularly arranged meetings in order to minimize their dissatisfaction, which could occur during the period where they worked from home.

By checking on the software developers well-being through regular online one-to-ones, the Submodule Owner was able to ensure that the software developers were doing well, and also made them feel recognized during the pandemic. These things led the software developers to not feel dissatisfied with the job supervision of the Submodule Owner and feel satisfied with the recognition they received, which motivated them. Without the online one-to-ones, it might have been difficult to check on the software developers, as they were unable to see each other physically and thereby share their opinions on their work.

5.2.4 Social activities

Besides the daily standup meetings, check-in meetings, and one-to-ones, the software developers also had online social activities. As mentioned in section 5.2, the social activities were included as they affected the software developer's social environment, which had an impact on their work performance. These activities consisted of Friday afternoon sessions which lasted a couple of hours and were facilitated by the Submodule Owner. During these online sessions, the Scrum team attended with several other people from the department to casually talk with each other (Appendix 2.2, 36:40-37:54). In addition, they could also drink a beer together online as well as play online games (Appendix 2.1, 31:16-32:18). Besides Friday afternoon sessions, the Scrum team also facilitated “online lunches”, where they ate lunch together online (Appendix 2.2, 53:46-55:06). The social activities can be categorized as clan control, since some of the activities were facilitated by the Scrum team themselves. Furthermore, the activities facilitated by the Submodule Owner were used to promote a set of common values within the team, which was to create good relations with each other, thus making it clan control. In the following quote by the Data Scientist, it can be seen that the social activities were positively perceived and much needed.

> *“I'm thinking, sometimes, whether we should be better at having lunch together online. I tried asking the others a few times, whether they wanted to have lunch, and they generally wanted. They were happy to have lunch together, but it's not something any of the others have asked for. Our department manager has arranged a few social get together events Friday afternoon, where we just talk casually together in the department. We’ve done that two times. So, that's also a good initiative.*"
>
> \- Data Scientist (Appendix 2.2, 36:40-37:54)

According to the Data Scientist, the Friday afternoon sessions were a good initiative, as it gave them the opportunity to get together and casually talk with each other. He also expressed that he would like to eat lunch together online more often with his colleagues, which showed that there was a need for more social activities within the Scrum team. Therefore, he took the initiative to ask his colleagues to eat lunch together, which they were happy to accommodate. While working from home, the Scrum team felt a need for more social interaction as there was a lack of social activities, where they could interact with each other outside of work. Consequently, the lack of social interaction negatively impacted their interpersonal relations. For instance, the Lead Financial Engineer mentioned that when they talked, as a team online, it was much more work related, and he missed the daily office talks that they had at the coffee machine (Appendix 2.3, 29:25-30:02). This shows that prior to the pandemic, the Scrum team had good interpersonal relations as they could be more open with each other and talk about things that were not work-related. Considering that the Scrum team had more social interaction with each other while working at the office, their interpersonal relations were better. However, during the pandemic, they had less social interaction with each other due to the lack of social activities, which resulted in a more formal relationship. This meant that their conversations revolved around work and were less personal compared to how it was prior to the pandemic. In addition, the Data Scientist also felt that there was a lack of social interaction with his colleagues while he was working from home, and thus tried to facilitate more social activities, such as online lunches. Even though his colleagues did not actively ask for it, they still felt it was much needed as they were happy to take part in the online lunches. The Submodule Owner also acknowledged that there was a lack of social activities, when they were working from home and planned to facilitate more in the future (Appendix 2.1, 31:16-32:18). Having more social activities was important for the Scrum team, as they could interact with each other more often, and thereby maintain their good interpersonal relations. This would happen naturally as they would have fun and enjoy the social activities together. However, since there was a lack of social activities during the pandemic, it prevented them in maintaining their good interpersonal relations with each other, which led to them being dissatisfied with the hygiene factor, interpersonal relations. This dissatisfaction had a negative impact on the software developers' basic need of interpersonal relations, which can hinder the possibility of fostering motivation. Besides the social activities having an effect on the software developers' motivation, it can also be seen that their desire for good interpersonal relations encouraged them to facilitate social activities and made the social activities facilitated by the Submodule Owner successful.

Due to the pandemic, the software developers felt a lack of social interaction with their colleagues, which prevented them in maintaining their good interpersonal relations. This led to dissatisfaction with the Scrum team's interpersonal relations, as the lack of social activities did not fulfill a basic need and could hinder the possibility of fostering motivation. Transaction was also evident in regard to the social activities, as their motivation for good interpersonal relations encouraged them to facilitate social activities, and made the social activities facilitated by the Submodule Owner successful.

5.3 Results

As seen in the findings, both WindCo's work process and performance management was highly influenced by SAFe, when the software developers were working from home during the pandemic.

Agile Work Process

The agile work process was mainly perceived positively by WindCo's software developers, as seen in figure 1, where they had three different control modes; behavior control, clan control, and self-control. The Sprint Planning and Sprint, respectively clan control and self-control, were useful during the pandemic, where the software developers were working from home. This was because these control enactments motivated them by enabling both shared and individual responsibility. The Sprint Planning motivated the software developers by allowing them to be responsible for the breakdown, arrangement, and delegation of tasks, which also made it possible to get the tasks of interest The Sprints motivated them by allowing them to be responsible for their own solutions to their tasks and time, which made their physical working conditions at home better. The success of enacting those controls were due to the software developers' shared and individual responsibility. The given responsibility made it possible for the software developers to perform the activities during the Sprint Planning and Sprints. Thus, the software developers must have a desire to be responsible in order for these control enactments to be successful. Furthermore, the behavior control, PI Planning sessions, were useful, as they helped the software developers to get a better understanding of the purpose of their work and make them feel a part of a greater community while they worked from home. This understanding of purpose and feeling of belongingness fulfilled one of the basic needs regarding the company policies and administrative practices. However, the PI Planning sessions also had a negative impact on the software developers' physical working conditions at home, as the software developers had to sit in front of the computer for two full workdays and have meetings, which was draining.

Performance Management

As illustrated in figure 1, within WindCo's performance management, the control enactments were also mainly perceived positively except for the lack of social activities. The control modes found in performance management were outcome control and clan control. The daily standup meetings were useful in the situation of working from home, as they could check up on each other every day and thereby get an overview of the Scrum team's progress, which fulfilled one of the basic needs regarding company policies and administrative practices. Also, the meetings motivated the software developers since they were given a shared responsibility, which also encouraged them to help each other, and thereby minimize stress deriving from the physical working conditions at home. In addition, the meetings made the software developers feel less disconnected when they were working from home, as they had these daily talks, which made their physical working conditions at home less dissatisfying. The daily standup meetings were conducted successfully due to the shared responsibility that encouraged them to help each other, which they did at these meetings. Furthermore, the check-in meetings and one-to-ones were also useful in a context of working from home. The check-in meetings motivated the software developers when they were working from home, as they had the opportunity to grow their skills by obtaining new knowledge from sharing knowledge with each other. The meetings were possible due to the software developers' eagerness to grow their skills and knowledge, which resulted in them facilitating the meetings. Thus, in order to successfully conduct check-in meetings, the software developers should have the desire to grow their skills and obtain new knowledge. In addition, the one-to-ones had a positive impact on the software developers since the Submodule

Owner wanted to ensure that the software developers were doing well during the pandemic. Thus, the one-to-ones helped fulfill the software developers' basic need regarding job supervision and made it possible to motivate the software developers by having the Submodule Owner recognizing the software developers' sentiments about working from home. Although the control enactments within performance management were mainly perceived positively, the software developers were not pleased with the lack of social activities while they were working from home. This lack of social activities prevented them in maintaining their good interpersonal relations, which they had prior to the pandemic. Thus, it resulted in the software developers' basic need of interpersonal relations not being fulfilled, which could hinder the possibility of fostering motivation. Transaction in this can be seen as the software developers' motivation for good interpersonal relations encouraged them to facilitate social activities and made the social activities facilitated by the Submodule Owner successful.

As the control enactments within SAFe were mainly perceived positively by the software developers, it showed that the implementation of SAFe worked well in the context of working from home during the pandemic.

Theme	Control	Hygiene & motivation factor
Agile work process	PI Planning sessions (behavior control)	Company policies and administrative practices - *no dissatisfaction* Physical working conditions - *dissatisfaction*
	Sprint Planning (clan control)	Responsibility - *satisfaction* Work itself - *satisfaction*
	Sprint (self-control)	Responsibility - *satisfaction* Physical working conditions - *less dissatisfaction*
Performance management	Daily standup meetings (outcome control)	Company policies and administrative practices - *no dissatisfaction* Responsibility - *satisfaction* Physical working conditions - *less dissatisfaction*
	Check-in meetings (clan control)	Growth opportunities - *satisfaction*
	One-to-ones (clan control)	Recognition - *satisfaction* Job supervision - *no dissatisfaction*
	Social activities (clan control)	Interpersonal relations - *dissatisfaction*

Figure 1: *Controls and hygiene/motivation factors*

5.3.1 Transaction Between Control and Motivation

The findings also showcased that the relationship between control and motivation was transactional, which argues for the awareness of the transactional relationship when controlling and motivating software developers during the pandemic. This is illustrated in figure 2, which shows different transactions between control modes and motivation/hygiene factors. In the figure, the relations between the controls and hygiene and motivation factors are visualized with arrows. If they point one direction it means the relation is causal, whereas multidirectional means it is transactional. The underlined hygiene and motivation factors are the ones that influence the control, thus making it relation transactional. Seven out of seven controls affected the motivation of the software developers, where the clan control enactments led to most motivation factors. An example of this was the one-to-ones, where the clan control influenced the software developers' motivation by having the Submodule Owner listen to the software developers' sentiments about their work, and thereby provide job supervision and recognition. On the other hand, five out of seven controls were affected by the software developers' motivation. The findings showed that it was mostly motivation factors that had an impact on the controls with the exception of a hygiene factor in the social activities. As seen in figure 2, the motivation factor, responsibility, affected most controls. For instance, the software developers must have a desire to grow in knowledge and skills in order for check-in meetings to be successful.

Theme	Control	Relation	Hygiene & motivation factor
Agile work process	PI Planning sessions (behavior control)	→	Company policies and administrative practices, physical working conditions
	Sprint Planning (clan control)	↔	Responsibility, work itself
	Sprint (self-control)	↔	Responsibility, physical working conditions
Performance management	Daily standup meetings (outcome control)	↔	Responsibility, company policies and administrative practices, physical working conditions, work itself
	Check-in meetings (clan control)	↔	Growth opportunities
	One-to-ones (clan control)	→	Recognition, job supervision
	Social activities (clan control)	↔	Interpersonal relations

Figure 2: *The transaction between control and motivation*

6. Discussion

In this chapter, we will discuss the findings found in the prior chapter. We will first present how our findings have contributed to the field of research presented in chapter 2. Next, we will provide the practical implications to help WindCo and other organizations that want to control and motivate their software developers who are working from home. Lastly, we present the limitations that we had encountered during this study.

6.1 Research Contributions

As mentioned earlier in chapter 2, based on our knowledge during this project period, there was no existing literature on how software developers can be motivated and controlled while working from home during the pandemic. Therefore, it is also challenging to conduct a comparison of our findings with findings from other studies that research the same subject. However, we see a possibility to compare with similar research, which was presented in chapter 2, by focusing on two main contributions derived from our results; (1) an agile method was useful during the pandemic and (2) the transactional relationship between control and motivation was evident. By comparing our study with other studies, we can find out how our research contributes to the existing literature, either being supportive, contradicting, or novel.

Agile Methods During the Pandemic

As mentioned previously in 5.3, we found that the control enactments within SAFe were mainly perceived positively by the software developers. For instance, during the Sprints, the software developers were able to be responsible for both their own solutions to their tasks but also the time. This showed that the implementation of SAFe worked well in the context of working from home during the pandemic, and thus, our findings suggest that agile activities within SAFe were beneficial for software developers working from home during the pandemic.

Previous research also suggests that implementing an agile method in the context of working from home during the pandemic is useful. This can be seen in the study by Schmidtner et al. (2021), who investigated the impact of the pandemic on the agile method. Their findings showcased that the agile method made it easier for the organizations to adapt to the changes caused by the pandemic (Schmidtner, Doering & Timinger, 2021, p. 10). This aligns with our findings, where we found the agile method to be beneficial for software developers working from home. Another study that can be compared to ours, is the study by Russo et al. (2020), which investigated the well-being and productivity of software professionals during the COVID-19 pandemic. It is comparable as it suggests that employees' need for autonomy needs to be fulfilled in order to ensure their well-being and productivity. Their findings showed that during the pandemic, their need for autonomy was not fulfilled which then had a negative impact on their motivation. Thus, in order to fulfill this need, the software professionals working from home must be able to freely structure, organize, and perform their tasks (Russo et al., 2020, p.28), which they can do through self-control. These findings are supported by us, as it can be seen that the software developers' need for autonomy was fulfilled during the Sprints where the individual software developers managed their own work schedule and formed their own solutions to their tasks. This led them to feeling responsible, and thereby motivated. In addition, this freedom enabled more flexibility for the software developers working from home, as they could manage their own work schedule. This was also deemed important in Butler and Jaffe's (2020) diary study of software engineers working from home during the COVID-19 pandemic. Their findings showed that the software engineers were mostly grateful for the flexibility they gained while working from home, as it gave them a better work life balance. Thus, they were able to structure their work according to their situation at home, which had an impact on their well-being at work (Butler & Jaffe, 2021, p. 11).

Control and Motivation Relation

As mentioned previously in 5.3.1, our findings showcased that the relationship between control and motivation was transactional. For instance, the software developers were able to make their own solutions to their task and manage their time, as they saw fit during the Sprints. As they were given this responsibility, it motivated them during the pandemic. However, it can also be seen the other way around, where the software developers' motivation for responsibility have led them to be able to perform the Sprints as they did. Thus, our study contributes to creating an awareness of the transactional relation between control and motivation when wanting to control and motivate software developers, who worked from home during the pandemic.

As mentioned previously in section 2.3, previous research only looked at the causal relationship where the controls influence the motivation of the employees. One of these studies is by van der Kolk et al. (2019), who investigated how the four control modes (behavior, outcome, clan, and self) had an influence on intrinsic and extrinsic motivation. This study is comparable to our study, since Herzberg's motivation-hygiene theory is based upon intrinsic and extrinsic motivation, as described in section 3.2, where the hygiene factors are extrinsic motivators and motivation factors are intrinsic motivators. Our findings support two confirmed hypotheses from their study, the first one being: "*self controls are positively associated with intrinsic motivation*". We agree with the explanation given by van der Kolk et al. (2019) that self-control plays a role in enabling employees to foster feelings of autonomy and competence, which lead to higher levels of intrinsic motivation (van der Kolk et al., 2019, p. 920). The software developers at WindCo were given autonomy to make independent decisions in order to solve their tasks during Sprints and manage their own schedule, which satisfied their needs for responsibility, thus increasing their intrinsic motivation. In addition, our findings support the second hypothesis: "*clan controls are positively associated with intrinsic motivation*". They added that this might have been due to clan control increasing the feeling of 'relatedness' among the employees by enhancing emotional ties to others and the overarching values and norms of the organization (van der Kolk et al., 2019, p. 920). Our findings do not point towards a feeling of relatedness, but instead a variety of motivation factors such as responsibility, work itself, growth opportunities, and recognition. This indicates that there is more complexity on how clan control is positively associated with intrinsic motivation than what the study argued for, as clan control satisfied the software developers emotional needs in more ways than one.

Another study that is supported by our findings and the findings of van der Kolk et al. (2019) is the study by Goldbach and Benlian (2015), which investigated how self-control and clan control affect third-party developers' intrinsic motivation (Goldbach & Benlian, 2015, p. 2). They found that both self-control and clan control had positive effects on third-party developers' intrinsic motivation (Goldbach & Benlian, 2015, p. 11). This was because the third-party developers had rated the self-control high as they largely agreed that they were able to self-manage the development process, set specific goals for their projects, define specific procedures for their project activities, and were given autonomy to set goals independently (Goldbach & Benlian, 2015, p. 9). Likewise, the developers rated clan control high as they largely agreed that they were actively trying to interact with other developers and understand the goals, values, and norms (Goldbach & Benlian, 2015, p. 9). Goldbach and Benlian found that the software developers were likely to have lower intrinsic motivation if they perceived decisions and activities less originated by themselves and also refused to interact and share common values and goals (Goldbach & Benlian, 2015, p. 13).

6.2 Practical Implications

Based on our findings, there are several things that organizations and managers need to be aware of in order to better control and motivate the software developers during the pandemic or when working from home. In this section, we will present the practical implications within the agile work process and performance management and refer to figure 3 for the overview of the implications.

In regard to the agile work process, our findings suggest that software developers should be allowed to be responsible for their work by letting them break down, arrange, and delegate tasks among themselves. Also, by allowing them to delegate the tasks, it gives the software developers the opportunity to work with the tasks of interest. As these things were possible at WindCo's Sprint Planning, it motivated the software developers during the pandemic. In addition, software developers should be allowed to form their own solutions to their tasks, and more or less be able to arrange their own time when working from home. This was possible in WindCo's Sprints, which motivated the software developers, and additionally improved their physical working conditions at home. Although the activities in Sprint Planning and Sprints were allowed, it should be noted that the software developers should also have a desire to be responsible in order for the activities to be successful. Furthermore, our findings suggest that managers should ensure that the software developers understand the purpose of their work from home, and can talk with other colleagues besides the ones in their own team. As this was possible in WindCo's PI Planning sessions, it fulfilled a basic need in regard to the company policies and administrative practices. Although the software developers were mainly motivated and pleased with the controls in the agile work process, our findings showcased that an improvement can be made in PI Planning sessions. During the PI Planning sessions, the software developers had meetings for two full workdays in front of the computer, which was draining and dissatisfied them in regard to the physical working conditions. In order to improve this, the facilitator of the PI Planning sessions, RTE, could notify the software developers when they would not be required to sit at their desk. This can, for instance, be when the different teams in the ART present their projects. Here, other teams, including the software developers, would be listeners, but could do other physical activities away from the screen in the meantime. An example could be going for a walk while having these meetings through the smartphone. In this way, meetings would be less draining for the software developers, as they would be able to activate themselves while also participating in the meetings. This idea can be supported by Haliburton and Schmidt, who stated that walking while having meetings is a way to introduce movement into busy work schedules, and furthermore mentioned that walking has shown to reduce stress and improve mental health and creativity (Haliburton & Schmidt, 2020, p. 2). Furthermore, the introduction of walking is also supported by Ricci et al. (2020), as it is one of the tips for home-based physical activity and sedentary behavior interruptions. Sedentary behavior typically includes “screen time”, such as using the computer, watching television, playing video games and so on. In this one tip, they state to take every chance to walk, like walking during a call (Ricci et al., 2020, p. 2). Thus, we suggest that organizations and managers should help with enabling software developers to be physically active whenever possible during the pandemic.

In relation to the performance management, managers need to ensure that their subordinates have regular meetings, where they can share how their work is progressing and receive help from others if they have troubles with their task. In WindCo, the Scrum team had daily standup meetings, which motivated them, as the meetings made them feel like everyone shared the responsibility of the tasks. Consequently, the software developers were willing to rearrange tasks among each other, which was necessary due to the circumstances of the pandemic. Managers need to be aware that without this shared sense of responsibility among the software developers it may be difficult to conduct the activities successfully. Furthermore, managers should facilitate or encourage their subordinates to facilitate meetings where the employees can share what they have learned while working on their individual tasks. This allows them to easily step in or help each other, as everyone has a general understanding of every task. The software developers in WindCo held check-in meetings, where they were able to share knowledge with each other about the tasks they have worked on. This motivated them, as they could grow their competencies and knowledge alongside the knowledge they obtained from working on their own tasks. However, in order for check-in meetings to be successful, the software developers must have an eagerness for growth in skills and knowledge. Besides that, managers should take initiative and set up one-to-one meetings with their subordinates to ensure their well-being, especially during the pandemic. These meetings can be an opportunity for the software developers to share their desires, worries, or problems they have in relation to their job or working from home. Managers should then take action accordingly in an effort to improve their well-being. Lastly, the software developers in WindCo felt like there was a lack of non-work related activities in the team, as the communication online was more formal than at the office. To solve this issue, managers should facilitate virtual social activities, these can vary in time and regularity, as long as the software developers have the chance to socialize during breaks or work. Alternatively, organizations can implement a system that simulates an office virtually. The idea is that if an employee is logged into the system, they are assigned to their own virtual office. The users can then set their virtual office to be open or closed. If open, it means the user in the office is connected to a video call which other people can connect to directly, simulating that they enter their virtual office. However, if the virtual office is set as closed, other colleagues can ask for permission to start a video call. Beyond working spaces, there can also be virtual common rooms where the employees can gather and socialize. This idea was inspired by Haliburton and Schmidt (2020), who argued that virtual offices can potentially encourage interaction between geographically distant workers. They added, it may recreate some of the dynamics of a real office, where coworkers can drop in on one another for short chats, to ask questions, or just to visit (Haliburton & Schmidt, 2020, p. 3).

Control	Motivational benefits	Recommendation
PI Planning sessions (behavior control)	• Understanding the purpose of their work • Feeling part of a bigger community	Ensure that employees understand the purpose of their work and enable them to be physically active when possible.
Sprint Planning (clan control)	• Getting responsibility of the tasks • Getting to work with interesting tasks	Ensure that employees are responsible and are allowed to break down, arrange, and delegate tasks among themselves, so that they can also get to work with the tasks of interest.
Sprint (self-control)	• Being responsible for their own tasks • Scheduling their own workdays	Ensure that employees are responsible, are allowed to form their own solutions to their tasks and arrange their own time.
Daily standup meetings (outcome control)	• Feeling of shared responsibility of the tasks • Employees helped each other	Ensure that employees have regular meetings, where they have opportunity to share their progression and get help from others.
Check-in meetings (clan control)	• Shared knowledge • Easier to step in for each other	Facilitate or encourage employees to facilitate meetings where employees share what they have learned while working on their tasks.
One-to-ones (clan control)	• Ensure well-being • Feeling of recognition • Acknowledging competent leadership	Set up individual meetings with the employees and take action accordingly to their desires or problems.
Social activities (clan control)	• Improve interpersonal relations	Facilitate regular virtual social activities, where employees can socialize outside of work, or implement a virtual office that can recreate office dynamics.

Figure 3: *Benefits and recommendations in relation to the control enactments*

6.3 Limitations

During this study, we came across some limitations. The first limitation was the use of Kirsch's (1997) control modes alone, which made it difficult to accurately categorize some of the control enactments within WindCo.

This was an issue we encountered when we tried to categorize the control enactment Sprint Planning, which could be categorized as both behaviour and clan control. It could be categorized as behaviour control, since Sprint Planning was part of the SAFe framework implemented by WindCo, and therefore according to Kirsch (1997), would be formal control. However, looking at how the Sprint Planning was perceived as clan control by the software developers, we chose to categorize it as such, even though it originally would be considered behavior control. As a result, the identified control enactments might leave room for small discrepancies. According to Cram and Wiener, control modes are not sufficient to fully understand the enacted control, and consequently supplemented it with the aspect of control style (Cram & Wiener, 2018, p. 715). In our case, by including control style it would have made it easier to categorize Sprint Planning, since formal controls also could be bilateral. The second limitation was in regard to the data collection, where we conducted a qualitative research study and gathered the empirical data through interviews. This had a few disadvantages. The first disadvantage was that the accuracy of the data relied on the interviewee's ability to accurately remember their experience of working from home during the pandemic. This can be problematic, as some interviewees might have found it difficult to recall their thoughts and experiences during that specific period, which could then lead to unreliable data (Alshenqeeti, 2014, p. 43). Therefore, it is recommended to supplement the interviews with other methods such as observation. This would have benefited us by giving us a better understanding and capturing the context in which the software developers worked in. Also, we would have become more open and discovery-oriented rather than relying and limiting ourselves to prior conceptualizations. Furthermore, we would have been able to discover things that the software developers might not have paid attention to or sensitive topics they might not have shared with us (Patton, 2002, p. 262). The second disadvantage of the data collection was the lack of information on the Scrum Master and the Product Owner, who were both perceived as controllers in this study. Initially, we interviewed the Submodule Owner, who was also regarded as a controller, and three software developers, who were the controllees. However, it was evident in the data that the software developers were managed daily by the Product Owner and had the Scrum Master to facilitate the Scrum activities. This meant that we might have missed valuable data that could have given us a better understanding of the context and the enacted controls.

7. Conclusion and Future Research

In this chapter, we will present the conclusion of the study and suggestions for further research on how managers can control and motivate software developers who are working from home during the pandemic.

7.1 Conclusion

This study investigated: "*How can managers motivate and control software developers who are working from home during the COVID-19 pandemic*". In order to answer the research question, we reviewed the existing literature in the field of control and motivation in ISD, which helped us identify the current research gaps as well as ascertain the relevance of the research question. The theoretical framework, which was based on Kirsch's control modes and Herzberg's motivation-hygiene theory, was used to analyze the empirical data gathered from the semi-structured interviews. The findings showed that the implementation of SAFe was beneficial for the software developers working from home during the pandemic. Within the agile work process, the software developers were motivated by the Sprint Planning and Sprint as these control enactments gave them both individual and shared responsibility, which according to Herzberg led to motivation. However, not all of the controls within the agile work process had a positive impact on the software developers working from home. This was the case with the PI Planning sessions, as it led to the software developers being dissatisfied with their physical working conditions. The PI Planning sessions helped the software developers to understand the purpose of their work and made them feel a part of a greater community, which led to them not being dissatisfied with the company policies and administrative practices. However, the software developers felt the sessions were draining to have, as they had to sit at their desk at home for two full workdays. This made them dissatisfied with the physical working conditions during the pandemic. In order to improve this, we suggested that the RTE notifies the participants whenever it is possible to have the meetings away from their desk. This enables the software developers to be physically active and thereby make the meetings less draining to have. In relation to the performance management, the daily standup meetings had positive effects on the software developers, as the meetings enabled them to check up on each other every day and thereby get an overview of the Scrum team's progress, which gave no dissatisfaction with company policies and administrative practices. Also, the daily standup meetings gave them the feeling of shared responsibility, which motivated them and also encouraged them to help each other. This made their physical working conditions less dissatisfying, as they were able to help each other when things at home became difficult at home. In addition, daily standup meetings also helped them in connecting with each other while they worked from home, which also made their physical working conditions less dissatisfying. Other controls within performance management that were not part of SAFe, such as one-to-ones and check-in meetings, also motivated the software developers. In the one-to-ones, the software developers felt motivated by the recognition they received, whereas, the check-in meetings enabled them to continuously grow their skills and knowledge while working from home. On the other hand, the software developers felt that there was a lack of social activities while working from home, which resulted in them being dissatisfied with their interpersonal relations, as they did not have much social interaction. To encourage social interaction among the software developers, managers can set up regular virtual social activities, where employees can chat during breaks or after work. Alternatively, organizations can implement a virtual office that can potentially recreate real office dynamics. Our findings also showed that there was a transactional relationship between control and motivation, thus, it is important to be aware of the transactional relationship when controlling and motivating software developers during the pandemic. Seven out of seven controls affected the motivation of the software developers, whereas, five out of seven controls were affected by the software developers' motivation. In order for both Sprint Planning and Sprints to be successful, the software developers needed to have a desire for responsibility. This was because they were the ones who had to break down, arrange, and delegate tasks, form the solutions for their own tasks, and manage their own work schedules. Similarly, responsibility was important in order to successfully conduct the daily standup meetings, as it enabled the software developers to offer help and step in for each other while they were working from home. Furthermore, the software developers must have an eagerness for growth in knowledge and skills, in order for them to facilitate the check-in meetings successfully. Lastly, due to the software developers' desire for social interaction, they tried to facilitate more social activities. Some of these findings were supported by existing studies about the agile method during the pandemic and the relation between control and motivation, which meant that our results contributed to the field of control and motivation in ISD.

7.2 Suggestions for Future Research

Further research is needed on how managers can control and motivate software developers who are working from home during the pandemic, as several aspects can have an influence on both control and motivation. One of the aspects is the cross-cultural differences in teams, where individual members have different cultural backgrounds. For instance, the Submodule Owner at WindCo mentioned that he had recently taken over a team in Portugal, and could definitely feel that there was a clear hierarchy between a manager and employee

(Appendix 2.1, 22:39-24:01). The cultural differences can have an impact on how certain controls are perceived and thereby also the software developers' motivation, considering there can be a language barrier or different work styles based on cultural norms. There are already several studies about cultural influences on collaborative work in software development teams, however, none of them takes control and motivation into account, which suggests that this research would be novel.

We believe that apart from looking into the cultural differences when controlling and motivating software developers who are working from home, future research could also investigate the impact on control and motivation by implementing a different framework than SAFe. Our results showcased that it was beneficial to use SAFe in the context of working from home during the pandemic, but the results might have been different if we had examined a team that had implemented a traditional development approach. The traditional approach might have been more advantageous or disadvantageous than an agile framework in terms of controlling and motivating software developers who are working from home. Furthermore, a comparative analysis can then be conducted to investigate which framework is most beneficial when working from home.

9. Bibliography

Alshenqeeti, H. (2014). Interviewing as a Data Collection Method: A Critical Review. *English Linguistics Research*, *3*(1).

Baddoo, N., Hall, T., & Jagielska-Hall, D. (2006). Software developer motivation in a high maturity company: A case study. *Software Process Improvement And Practice*, *11*(3).

Bartik, A. W., Cullen, Z., Glaeser, E. L., Luca, M., & Stanton, C. (2020). *What Jobs are Being Done at Home During the COVID-19 Crisis? Evidence from Firm-Level Surveys*. Massachusetts: Business Harvard School.

Beecham, S., Baddoo, N., Hall, T., Robinson, H., & Sharp, H. (2008). Motivation in Software Engineering: A systematic literature review. *Information And Software Technology*, *50*(8-9).

Beecham, S. (2014). Motivating Software Engineers Working in Virtual Teams Across the Globe. In G. Ruhe & C. Wohlin, *Software Project Management in a Changing World*. Berlin: Springer.

Butler, J., & Jaffe, S. (2021). *Challenges and Gratitude: A Diary Study of Software Engineers Working From Home During Covid-19 Pandemic*. Retrieved from https://ieeexplore.ieee.org/abstract/document/9402021

Campbell, J. L., Quincy, C., Osserman, J., & Pedersen, O. K. (2013). Coding In-depth Semistructured Interviews: Problems of Unitization and Intercoder Reliability and Agreement. *Sociological Methods & Research*, *42*(3).

Chege, I. (2009). *Intelligent Verbatim Transcription*. Retrieved from https://weloty.com/intelligent-verbatim-transcription/

Cram, A. W., & Wiener, M. (2018). Perceptions of control legitimacy in information systems development. *Information Technology & People*, *31*(3).

Dartey-Baah, K., & Harlley, A. (2010). Job Satisfaction and Motivation: Understanding its impact on employee commitment and organisational performance. *Academic Leadership: The Online Journal*, *8*(4).
Downe-Wamboldt, B. (1992). Content analysis: Method, Applications, and Issues. *Health Care for Women International*, *13*(3).

Elkjaer, B., & Simpson, B. (2011). Pragmatism: A Lived and Living Philosophy. What Can It Offer to Contemporary Organization Theory?. *Research In The Sociology Of Organizations*, *32*.

Farndale, E. (2016). Performance Management. In A. Wilkinson & S. Johnstone, *Encyclopedia of Human Resource Management*. Massachusetts: Edward Elgar Publishing.

Flyvbjerg, B. (2006). Five Misunderstandings About Case-Study Research. *Qualitative Inquiry*, *12*(2).

Ford, D., Storey, M., Zimmerman, T., Bird, C., Jaffe, S., Maddila, C., Butler, J. L., Houck, B., & Nagappan, N. (2020). *A Tale of Two Cities: Software Developers Working from Home During the COVID-19 Pandemic*. Retrieved from https://arxiv.org/abs/2008.11147

França, A. C. C., de Araújo, A. C. M. L., & da Silva, F. Q. B. (2013). Motivation of Software Engineers: A Qualitative Case Study of a Research and Development Organisation*. 2013 6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*.

Fusch, P. I., & Ness, L. R. (2015). Are We There Yet? Data Saturation in Qualitative Research. *The Qualitative Report*, *20*(9).

Goldbach, T. , & Benlian, A. (2015). Understanding Informal Control Modes on Software Platforms – The Mediating Role of Third-Party Developers' Intrinsic Motivation. *Thirty Sixth International Conference on Information Systems*.

Haliburton, L., & Schmidt, A. (2020). Move, Collaborate, and Iterate: Improving the Work from Home Experience. *Virtual Symposium On The New Future Of Work*.

Hennink, M., & Kaiser, B. (2019). Saturation in Qualitative Research. In P. Atkinson, S. Delamont, A. Cernat, J.W. Sakshaug, & R.A. Williams (Eds.), *SAGE Research Methods Foundations.* Thousand Oaks: SAGE Publications.

Herzberg, F. (1987). One More Time: How Do You Motivate Employees?. *Harvard Business Review*, *81*(1).

Herzberg, F. (1993). *Motivation to Work*. New York: Routledge.

Hsieh, H., & Shannon, S. E. (2005). Three Approaches to Qualitative Content Analysis. *Qualitative Health Research*, *15*(9).

Huang Chua, C. E., & Myers, M. D. (2018). Social Control in Information Systems Development: A Negotiated Order Perspective. *Journal Of Information Technology*, *33*(3).

Jacobs, R.L. (2019). *Work Analysis in the Knowledge Economy: Documenting What People Do in the Workplace for Human Resource Development*. Cham: Springer Nature.

Johnson, R. E., Muraven, M., Donaldson, T. L., & Lin, S. (2018). Self-Control in Work Organizations. In D. L. Ferris, R. E. Johnson & C. Sedikides, *The Self at Work: Fundamental Theory and Research* (1st ed.). New York: Routledge.

Kanfer, R., Chen, G., & Pritchard, R. D. (2008). *Work Motivation: Past, Present and Future*. New York: Routledge.

Kaplan, J., Frias, L., & McFall-Johnsen, M., (2020). *A third of the global population is on coronavirus lockdown - here's our constantly updated list of countries and restrictions.* Retrieved from https://www.businessinsider.com/countries-on-lockdown-coronavirus-italy-2020-3?op=1&r=US&IR=T

Kirsch, L. J. (1996). The Management of Complex Tasks in Organizations: Controlling the Systems Development Process. *Organization Science*, *7*(1).

Kirsch, L. J. (1997). Portfolios of Control Modes and IS Project Management. *Information Systems Research*, *8*(3).

Kvale, S., & Brinkmann, S. (1996). *InterViews: An Introduction to Qualitative Research Interviewing* (1st ed.). Thousand Oaks: Sage Publications

Kvale, S., & Brinkmann, S. (2009). *InterViews: Learning the Craft of Qualitative Research Interviewing* (2nd ed.). Thousand Oaks: SAGE Publications.

University of Pittsburgh. (n.d.). *Laurie J. Kirsch*. Retrieved from https://www.business.pitt.edu/people/laurie-kirsch

Lahdjel, A., Mohamoud, H., Rasmussen, L., Huynh, L., & Untaru, M. (2021). *A Comparative Case Study Of Software Developers' Motivation During the COVID-19 Lockdown* (Postgraduate). Aalborg University. Retrieved from https://drive.google.com/file/d/1vT0Nq1ZzdRDEBucfwfW7_0c2EpjCrLJ0/view?usp=sharing

Lazar, J., Feng, J. H., & Hochheiser, H. (2017). *Research Methods in Human-Computer Interaction* (2nd ed.). Cambridge: Morgan Kaufmann.

Legault, L. (2016). Intrinsic and Extrinsic Motivation. In V. Ziegler-Hill & T. Shackelford, *Encyclopedia of Personality and Individual Differences*. Cham: Springer.

Lombard, M., Snyder-Duch, J., & Bracken, C. C. (2002). Content Analysis in Mass Communication: Assessment and Reporting of Intercoder Reliability. *Human Communication Research*, *28*(4).

Mills, A. J., Durepos, G., & Wiebe, E. (2010). Explanatory Case Study. In *Encyclopedia of Case Study Research*. Thousand Oaks: SAGE Publications.

Orlikowski, W. J. (1991). Integrated information environment or matrix of control? The contradictory implications of information technology. *Accounting, Management & Information Technology, 1*(1).

Ozguner, Z., & Ozguner, M. (2014). A Managerial Point of View on the Relationship Between Maslow's Hierarchy of Needs and Herzberg's Dual Factor Theory. *International Journal of Business and Social Science, 5(7).*

Padalay, P. (2020). *What's the Difference Between Scrum and SAFe?*. Retrieved from https://agilemania.com/blog/scrum-vs-safe/

Patricia, O. (2015). Improving Interpersonal Relationship in Workplaces. *Journal Of Research & Method In Education*, *5*(6).

Patton, M. Q. (2002). *Qualitative Evaluation and Research Methods* (3rd ed.). Thousand Oaks: SAGE Publications.

Patton, M. Q. (2015). *Qualitative Evaluation and Research Methods* (4th ed.). Thousand Oaks: SAGE Publications.

Quain, S. (2018). *Traditional Vs. Contemporary Organizational Structure: Formal & Informal*. Retrieved from https://smallbusiness.chron.com/traditional-vs-contemporary-organizational-structure-60243.html

Ralph, P., Baltes, S., Adisaputri, G., Torkar, R., Kovalenko, V., & Kalinowski, M. et al. (2020). Pandemic programming: How COVID-19 affects software developers and how their organizations can help. *Empirical Software Engineering*, *25*.

Regeringen. (2020). *Situationen kommer til at stille kæmpe krav til os alle sammen.* Retrieved from https://www.regeringen.dk/nyheder/2020/statsminister-mette-frederiksens-indledning-paa-pressemoede-i-statsministeriet-om-corona-virus-den-11-marts-2020/

Remus, U., Wiener, M., Saunders, C., Mähring, M., & Kofler, K. (2016). Control Modes Versus Control Styles: Investigating ISD Project Control Effects at the Individual Level*. International Conference on Information Systems (ICIS)*.

Remus, U., Wiener, M., Saunders, C., & Mähring, M. (2020). The Impact of Control Styles and Control Modes on Individual-level Outcomes: A First Test of the Integrated IS Project Control Theory. *European Journal of Information Systems*, *29*(2).

Ricci, F., Izzicupo, P., Moscucci, F., Sciomer, S., Maffei, S., & Di Baldassarre, A. et al. (2020). Recommendations for Physical Inactivity and Sedentary Behavior During the Coronavirus Disease (COVID-19) Pandemic. *Frontiers In Public Health*, *8*(199).

Roberts, J. A., Hann, I., & Slaughter, S. A. (2006). Understanding the motivations, participation, and performance of open source software developers: A longitudinal study of the Apache projects. *Management Science, 52*(7).

Russo, D., Hanel, P., Altnickel, S., & van Berkel, N. (2021). Predictors of Well-being and Productivity of Software Professionals during the COVID-19 Pandemic – A Longitudinal Study. *Empirical Software Engineering*, *26*.

Santana, M., & Robey, D. (1995). Perceptions of control during systems development: Effects on job satisfaction of systems professionals. *Computer Personnel, 16*(1).

Scaled Agile. (2021a). *PI Planning*. Retrieved from https://www.scaledagileframework.com/pi-planning/

Scaled Agile. (2021b). *Release Train Engineer and Solution Train Engineer*. Retrieved from https://www.scaledagileframework.com/release-train-engineer-and-solution-train-engineer/

Schmidtner, M., Doering, C., & Timinger, H. (2021). *Agile Working during COVID-19 Pandemic*. Retrieved from https://ieeexplore.ieee.org/document/9392229

Schrage, Z. (2021). How Can Software Development Teams Be Controlled During the COVID-19 Pandemic. In C. Hovestadt, J. Recker, J. Richter & K. Werder, *Digital Responses to Covid-19* (1st ed.). Cham: Springer.

Schwaber, K., & Sutherland, J. (2020). *Scrum Guide*.

Sigal, T. (2016). *The CCM Maturity Model: where do you fit?*. Retrieved from https://www.quadient.com/blog/ccm-maturity-model-where-do-you-fit

Theresia, L., Lahuddin, A. H., Ranti, G., & Bangun, R. (2018). The Influence of Culture, Job Satisfaction and Motivation on the Performance Lecturer / Employees. *Proceedings of the International Conference on Industrial Engineering and Operations Management*.

Trate, J. M., Teichert, M., Murphy, K. L., Srinivasan, S., Luxford, C. J., & Schneider, J. L. (2020). Remote Interview Methods in Chemical Education Research. *Journal Of Chemical Education, 97*(9).

van der Kolk, B., van Veen-Dirks, P. M. G., & ter Bogt, H. J. (2019). The Impact of Management Control on Employee Motivation and Performance in the Public Sector. *European Accounting Review*, *28*(5).

Wells, D. (1999). *When Should Extreme Programming be Used?*. Retrieved from http://www.extremeprogramming.org/when.html

WHO. (2020a). *Coronavirus disease (COVID-19)*. Retrieved from https://www.who.int/emergencies/diseases/novel-coronavirus-2019/question-and-answers-hub/q-a-detail/coronavirus-disease-covid-19

WHO. (2020b). *WHO Announces COVID-19 Outbreak a Pandemic.* Retrieved from https://www.euro.who.int/en/health-topics/health-emergencies/coronavirus-covid-19/news/news/2020/3/who-announces-covid-19-outbreak-a-pandemic

WHO. (2021). *Coronavirus*. Retrieved from https://www.who.int/health-topics/coronavirus#tab=tab_1