UNIVERSITY OF GOTHENBURG
SCHOOL OF BUSINESS, ECONOMICS AND LAW

Managing Software Development Efficiency

A Swedish experience of applying agile method at telia.se

Industrial and Financial management
Bachelor thesis
Spring 2015

Authors
Lund, Emma 930320
Nyman, Sarah 920820

Tutor
Mansouri, Ziaeddin

Acknowledgements

We would like to send our sincerest appreciation to all employees at TeliaSonera for their contribution to this thesis and their warm welcoming and cooperation. An extra thanks to Mikael Eriksson for all the help and guidance during this whole process.

Finally, we would like to extend a big thank you to our supervisor Zia Mansouri for the feedback and the all his help to navigate us through the difficulties of writing this thesis.

Emma Lund

Sarah Nyman

Gothenburg 2015-05-28

Abstract

Background and problem: Due to rapid changing market conditions in the IT industry, companies within the software industry have to respond to constantly changing customer needs. Some organizations have implemented an agile approach to manage this difficulty. Many small and middle-sized firms have successfully implemented the agile methodology in software development, however larger organizations can face some challenges when implementing agile approach. Telia.se has recently implemented agile method in their software development and even though the approach implies positive changes in the software development efficiency, this field of study has lack of empirical evidence and need to be researched.

Purpose: The purpose of this study is to gain better insight into agile software development and its software development efficiency at telia.se. This is carried out by a case study at telia.se where different criterions are examined in order to try to measure what the implementation of the agile approach has led to.

Method: A case-study setting was chosen as research design. The empirical material has been collected through qualitative, semi-structured interviews with eight employees at telia.se. All interviews were made face-to-face. Afterwards, all material has been analyzed and compared to the theoretical background in order to be able to answer our research question.

Results and conclusion: In this case study, the collected empirical material showed that the software development efficiency has increased since the implementation of agile methods at telia.se and permanent teams are one of the biggest contributions to this. There are several challenges and difficulties to implement agile software development in a larger organization, which must be taken into account.

Key words: Agile, Scrum, Waterfall, Efficiency, Agile software development, Software development efficiency

Glossary

Agile - A firms' constant readiness to create and embrace change through its internal resources and relationships with its environment in order to respond to changing business environment.

Agile software development - A working method within software development which uses iterative development processes and frequent releases of developed functionality.

Scrum - Scrum is an iterative and incremental agile software development methodology for managing product development.

Waterfall - A classic working method within software development.

Functionality - Developed solutions to the website.

Output - Features of developed functionality.

Outcome - The effects and impact of developed functionality.

List of figures

List of tables

Table of content

1. Introduction

This chapter provides a background to agile software development with a subsequent problem discussion, in which the topical problems of this context will be discussed. The research question is created and the purpose of the study is clarified in order to give the reader a clear view on how this study will research agile software development.

1.1 Background

In today's society, companies are exposed to uncertain business nature due to globalization and volatile demand. Thus, it is crucial that companies respond to rapid changes in market conditions to be able to survive (Moniruzzaman & Hossain, 2013). This is especially important within the IT industry, as it suffers extra from changing market conditions (Torrecilla-Salinas, Sedeño, Escalona & Mejías, 2015, Börjesson & Mathiassen, 2005) In order to manage this difficulty, many IT organizations have implemented an agile approach in their software development (Börjesson & Mathiassen, 2005; Abrahamsson, Salo & Ronkainen, 2002). This means that you work in changing processes in order to respond to market changes in a faster pace (Moniruzzaman & Hossain, 2013). Goldman, Nagel and Preiss (1995) chooses the definition ”A continuous readiness to change” to describe agile working methods.

Software development has traditionally been executed in the waterfall working method, in which they work in long projects with a start and expiration date with specified customer requirements throughout the whole development process (Dubey, Jain & Mantri, 2015, Torrecilla-Salinas et al., 2015). This requires an stable and recognizable business environment, which however do not exist in reality. As a result, developed IT functionality within this method are often outmoded when reaching the market (Jalote & Kurien, 2004) As a reaction to this problem, agile working methods were established during the 1990s in order to develop more updated functionaliy to meet customer demands (ibid). Agile methods facilitate the development process when facing dynamic changing business requirements (Hong, Thong, Chasalow and Dhillon, 2011). The development of agile approaches had its boom due to the rampage of internet, as this created entirely different market conditions with increased competition between firms (Torrecilla-Salinas et al., 2015; Abrahamsson et al., 2002; Thong et al., 2011; Lee & Xia, 2010). Increased competition also sets pressure to companies to work as efficient as possible, as companies compete with scarce resources.

Even though the first agile approaches in software development was established during the 1990s, it is not until recent years agile methods have been commercialized and commonly used (Torrecilla-Salinas et al., 2015). Many small and middle sized firms have successfully implemented the agile software development. Several large organizations have implemented agile software development as well, however, these processes have been more challenging (Leffingwell, 2007, 2010; Krebs, 2008 Vähäniitty, 2012). Telia.se is one business unit within the corporation TeliaSonera AB, which officially stated around one year ago that they worked with agile software development. This implementation of agile method is thereby one of the first made in the context of a large IT organization in Sweden.

Today "Scrum" is one of the most widely used methods within agile software development (Barlow et. al 2011; Vlietland & Vliet, 2014). This method is also implemented at telia.se in order to receive an agile approach in their development process. In this working method, employees are working in permanent teams instead of projects. The team's work in shorter iterations called Sprints in which they deliver required functionalities called User stories. The Product owner gives requirements in elaboration of the functionality in consideration to the stakeholders, where the customer is the most important. Each team has a Scrum master, which is responsible for the output. Developed functionality must meet stakeholders demand; otherwise it will be useless and consequently rejected as solution to the website (Torrecilla-Salinas et al., 2015).

1.2 Pre-study

Before this study was initialized, a pre-study was made at telia.se in order to receive a deeper understanding about what the most problematic questions are in the field of investigation, and thereby what the purpose of the study could be. It was presented to us that telia.se, which is a part of TeliaSonera AB, recently had begun to work with agile methods. Telia.se is among one of the first large organization in Sweden to implement this working method.

Further, we received information about how telia.se works with agile methods. Accordingly, we perceived that the issue of efficiency in software development activities was an ambiguous concern for telia.se because of adopting the new method of agile approach.

1.3 Problem discussion

There is a discussion whether agile software development is beneficial or not. Some organizations have successfully adopted the agile methodology in software development, and thereby become more competitive and efficient with reduced costs. Other organizations have been skeptical to agile software development, arguing that it will make work more complex and not turn out advantageous (Barlow et. al 2011). Critics argue that cost of agile development will be greater than the possible benefits (Ambler, 2008; Rising and Janoff, 2000; Selic, 2009). Cockburn (2001) argues that agility can be hard to achieve in practice. Few organizations are able to keep up with business demand and efficiently handle business needs (Koch, 2006), which result in substantial financial loss (Austin & Devin 2003).

The deployment of agile methods is considered to be one of the main challenges of agile software development (Laanti, Salo & Abrahamsson, 2010). This can partly be explained by organizational culture and its interpretation of agile. An agile adoption can be seen as an organizational culture change, which is one of the most challenging issues to perform in an organization (Dybå & Dingsöyr 2010). There is an additional challenge for large organizations to integrate agile methods since they often already have existing standards and business processes within complex systems (Barlow et. al 2011; Leffingwell, 2007, 2010; Krebs, 2008; Vähäniitty, 2012). It is facilitating if the transformation process is in consensus with the old system, as it otherwise could lead to misunderstandings which decreases an efficient work (Trendowicz & Münch, 2009). Furthermore, to be efficient in agile software development, it is necessary with a simple and decentralized organizational structure (Lee & Xia, 2010). This can be challenging in large organizations, which often have centralized structures and therefore need to change governance approach (Stettinan & Hörz, 2014)

Earlier studies show that the lesser experience you have from agile software development, the more negative attitude towards agile methods. On the hand, the more experience from agile methods, the more positive opinions. However, the possible negative attitude towards agile in the adoption process might prevent employees to perform to their fullest potential, which could directly affect firm's efficiency (Laanti et. al 2010).

There is a problem with defining requirements on the functionalities in agile software development. Especially in large organizations that might have standardized processes and communication steps due to the many interdependencies. The standardized processes make communication less expensive (Barlow et al., 2011). Although in large organizations the functionality of IT will be achieved through a chain of interdependent software applications (Vlietland & Vliet, 2014), it is not easy to manage changing business requirements within such chains. As Vietland and Vliet (2014) argue, one remedy for managing this problem is to employ agile methods such as Scrum. In the agile method Scrum that is implemented at telia.se, collaboration and communication are essential in order to deliver functionality and consequently cannot be standardized (Vlietland & Vliet, 2014). Instead, agile methods might lead to lack of documenation and discipline, which can result in loss of well-organized history and experience (Boehm & Turner, 2004). Boehm and Turner (2004) argue that organizations require both agility and documentation and discipline in order to be successful.

Agile method is a new and trendy working method with little empirical evidence (Barlow et. al 2011, Lee & Xia, 2010; Abrahamsson et al., 2002; Hong et al., 2011; Vähäniitty, 2012). Many organizations yet are considering using agile practices (Abrahamsson Warsta, Siponen &

Ronkainen). Since software development has a strong practical orientation, it is argued that this needs to be studied in real life situations in form of empirical studies (Abrahamsson et. al., 2003). Thus, this study aims to fill this gap in literature.

1.4 Research question

Considering the problem discussion established above, this study will target the following question:

How has an agile method in software development activities improved the software development efficiency of telia.se?

1.5 Purpose of the study

The purpose of this study is to gain better insights into agile software development by investigating in the field of telia.se and making sense of whether the adoption of an agile method has improved software development efficiency.

2. Theoretical framework

This chapter will give an explanation of the term software development efficiency in order to provide the reader with a greater understanding in how this can be measured in this study's context. Furthermore will the earlier working method waterfall and the current established method agile software development be reviewed more deeply to provide the reader with greater understanding of the empirical findings and analysis.

2.1 Conceptualizing Software development efficiency

Efficiency is a vague concept that needs to be conceptualized in the context of software development. Consequently, the phrase that will be used in order to measure efficiency in this study is software development efficiency. Though, it can be hard to measure software development efficiency, as software development is a human-based activity with much unpredictability from the outset. This is problematic when trying to define an reliable definition of efficiency (Melo, Cruzes, Kon & Conradi, 2012).

Efficiency derives to the relationship between the efforts of the business and the results of the business (NE.se, 2015). Efficiency can be correlated to March (1991) expression “exploitation”, which means that the firm refines old certainties in order to improve already existing resources. In other words, efficiency can be derived from creating better prerequisites in order to improve usage and development of already existing business activities (March 1991). According to March (1991), efficiency is created when making refinements in production, implementation and execution. It is argued that efficiency is essential for organizations, as they compete with scarce resources. However, it is further argued that the outcome of exploitation depends on the timing and distribution within the organization (ibid). Efficiency can also be derived from certainty, speed, proximity and clarity of feedback (ibid). Furthermore, efficiency is seen as the refinement and extension of existing competences and technologies.

According to Frökjer, Hertzum and Hornbaek (2000), efficiency can be derived from completion time, effectiveness in form of goal achievement, quality of functionalities and lesser error rates. Furthermore, it is said that efficiency can be derived from the accuracy and completeness in combination with resources expended in achieving them. Time-to-market and learning time are also considered as measurements on efficiency (Frökjer et al, 2000). Good performances, that are a consequence from well performed execution of a sequence of actions, are also considered to yield efficiency (ibid).

To summarize everything mentioned above, software development efficiency somehow measures how efficiently an organization develops functionalities in order to satisfy customers (Clutterbuck, Rowlands & Seamons 2009). According to Melo (2012), software development efficiency in an agile approach can be measured by using several criterions such as personnel satisfaction, learning process, customer satisfaction, quality on the functionalities, time-to-market, communication, flexibility and team productivity. In this study, these criterions will be used in order to investigate the research question.

2.1.1 Personnel satisfaction

Team capabilities and skills among personnel are the most important aspects that influence software efficiency (Maxwell & Forselius, 2000; Tan, Li, Boehm, Yang, He & Moazeni, 2009). Agile development has a people-centric approach and recognizes the employees as the most value bringing in the development process (Nerur & Balijepally, 2007; Lee & Xia, 2010). In software development, it is crucial to work with personnel with the right skills and empowering them in order to gain efficiency (Chow & Cao, 2008; Highsmith, 2004). Consequently, satisfied personnel are crucial in order to create an efficient work.

2.1.2 Learning process

Better learning process among employees is created when establishing an improved learning environment. In this context, a better learning process is created when the learning among people, software processes, functionality and working tools has enhanced (Quemer &

Henderson-Sellers, 2008). Learning practices enable companies to respond to changes in their environment (Dove, 2001) which facilitate an efficient work. Jin-Hai (2003) suggests that procession learning and "learning by doing" should be increased in an agile working method. Frökjer et al. (2000) explicitly states that learning could be seen as an indicator on efficiency. March (1991) expression exploation, which was mentioned earlier, is derived from improving old certainties in the organizational learning. Moreover, the outcome of an efficient work leads to better knowledge making among employees (Clutterbuck, Rowlands & Seamons 2009). Therefore increased knowledge among employees should be seen as an indicator on efficient work.

2.1.3 Customer satisfaction

Customer satisfaction is the user's comfort and positive attitudes against new developed functionality (Frökjer, Hertzum & Hornbaek 2000). Customer satisfaction can in this context be defined more explicitly as "users' overall experience of new implemented information systems" (Hong et al., 2011), which can be seen as the outcome of an efficient work (Ramirez & Nembhard, 2009; Melo et al., 2012). This means that work within the company is efficient in order to deliver products that satisfy customers. However in information systems research studies, it is important to consider that the historical context of the company can affect satisfaction levels (Orlikowski & Baroudi, 1991). An agile approach aim to satisfy the customer through continuous deliverable of implemented features due to the work in shorter iterations (Garg, 2009). It can be tricky to satisfy customer needs in the IT industry, as they continuously are changing (Lee & Xia, 2010).

2.1.4 Quality on the functionalities

Quality of developed functionalities involves better usability and better fitting to the task environment (Atkinson, 1999; Quemer & Henderson-Sellers, 2008). In software development, many argues that it is better to carry out integration of developed functionalities regularly in order ta maintain better quality (Bhoola & Mallik, 2014). According to Conboy and Fitzgerald (2004), quality is increased when developing simplistic and economical functionalities.

Clutterbuck, Rowlands and Seamons (2009) argues that continuity in design with increased quality, can be seen as indicator of better efficiency, as a consequence of better work practices. Earlier detections of bugs are also seen as an outcome of a more efficient work, which could be derived from better quality of functionalities. Efficiency regarding quality can also be derived from whether the solution is the right deliverable for both customer and developer. Quality issues of developed functionalities emerge with bad timing and insufficient testing of the developed functionality (Bjarnson, Wnuk & Regnell, 2012).

2.1.5 Time-to-market

Time-to-market measures the time it takes from a customer demand occurs, till the customer can take part of the developed functionality. It is important to minimize time-to-market in the IT industry due to the rapidly changing business environment (Börjesson & Mathiassen, 2005). According to Frökjer et al. (2000), an indicator of efficiency is explicit minimized task completion time. Moreover, they state faster task completion time as the primary indicator on an efficient development process. An efficient work leads to shorter time-to-market when developing functionalities in software development (Melo et al., 2012), which includes task completion time. According to March (1991), speed is a proof on better efficiency.

2.1.6 Communication

It is important with smooth communication in a development process in order to be as efficient as possible. Especially in an agile method, which requires collaboration between different stakeholders (Heikkilä, Paasivaara, Rautiainen, Lassenius, Toivola & Järvinen, 2014). It is argued that communication is crucial in order to ensure that coordination is satisfying (Vlietland & Vliet, 2014). However, due to the high complexity in IT landscapes, communication can be struggling (Heikkilä et al., 2014). Interactions among team members and interactions with other teams, customers, and suppliers directly affect team efficiency in agile approach (Melo et al, 2012). Clutterbuck, Rowlands and Seamons (2009) state that improved quality of interaction should lead to better efficiency when implementing an agile approach. Furthermore, less misunderstandings, which could be derived from better communication, is something that indicates better efficiency (Trendowicz and Münch (2009). According to Frökjer et al. (2000), efficiency can in an software development process, be measured by the level of communication.

2.1.7 Flexibility

Flexibility in software development can be seen as to which extent the working method handle expected or unexpected changes (Quemer & Henderson-Sellers, 2007). According to Conboy (2009), flexibility can be derived to a modification or change of existing capability as a consequence of adaptability to market changes. In a software development context, flexibility can be seen as the ability of a working method to create change, or proactively and reactively facilitate change within reasonable time with internal resources and relationship with the environment" (Conboy, 2009). Furthermore, Golden and Powell (2000) describe flexibility within software development as either "length of time it takes for an organization to respond to environmental change" or "adapt within a given time frame". According to Melo et al. (2012), an increased flexibility leads to a more efficient work in an agile approach. Furthermore, Clutterbuck, Rowlands and Seamons (2009) argue that flexibility leads to better efficiencies among personnel.

2.1.8 Team productivity

Productivity is the total output per unit effort (Jalote & Kurien, 2004). In an agile software development context, productivity can be measured by numbers of completed tasks per iteration (Svensson, 2005). However, as many companies measure productivity in different ways (Melo et al., 2012), the team members' own perception of productivity will be used in this study in order to measure productivity. Team productivity is one key issue in IT organizations (Melo et al., 2012). Employee's turnover, team composition and resource allocation are seen as the biggest influences to the team productivity in an agile method (Melo et al., 2012). Team performance and productivity depend on efficient and good execution of several actions, which in yields stable results (Trendowicz and Münch, 2009).

2.2 Waterfall method

Waterfall model is a traditional in software development (Dubey et. al, 2015). It was developed during the 1970's to overcome the problems when performing large software development projects (Boehm 1976; Royce 1970). Waterfall model is a sequential model,which means that it has different phases that are executed one by one in a downstream flow. Each phase has clearly defined deliverables and is accomplished in a specified period of time, so that the subsequent phase is initiating when the previous phase is completed (Jalote, Palit & Kurien, 2004).

Waterfall is best suited to use when the description of what the final product will be is clear. Furthermore, it is also suitable to use when characterization of the functionality is more important than speed (Dubey et. al, 2015) and when project- and team sizes are larger. However, there are challenges in the waterfall model that have been identified in empirical studies (Petersen & Wohlin, 2010). The biggest drawback is the assumption of a stable and known reality, which do not exist in reality (Jalote et. al, 2004). Thus, management is one of the areas of concern, since requirements are very hard to manage. Consequently, this has been identified as the main reason for failure (cf. Thomas 2001; Jarzombek 1999; Johnson 2002). Customers' needs might not be addressed until the end of the project (Jarzombek 1999), resulting in that much of the functionality is not implemented (Johnson 2002). Furthermore, it is complicated to finally test and integrate the overall system (Jones, 1995). Studies show that only a small portion of the developed functionality actually has been deployed or used. This is due to the changing needs and the lack of opportunity to clarify misunderstandings. This depends mostly on poor feedback on the functionality, which is a consequence of the difficulties for customers to actual provide this (Cohen et al. 2001).

2.3 Agile method

Agile software development is an iterative and incremental approach to software development, which is performed through cooperation and independent teams (Moniruzzaman & Hossain, 2003). Work is done in shorter cycles, thus allowing building of functionality in smaller steps (Miller 2001). The purpose of the approach is to produce high quality solutions in a cost effective and timely manner, in order to meet the changing needs of its stakeholders (Moniruzzaman & Hossain, 2003). Quality of the functionalities can be measured in terms of process results and is determined by "how effectively a process meets the customer's needs" (Schneiderman, 1996). According to (Raschke, 2010), functionalities quality can be reflected by customer satisfaction.

2.3.1 The overall aspects of agile method

In today's business environment, organizations are constantly changing their software requirements in order to adjust into a new and dynamic environment (Moniruzzaman & Hossain, 2003). Since the globalization of the world and the rise of Internet, the customer needs are in a constant change. Therefore it is crucial for corporations to be able to adapt to changing customer needs (Torrecilla et al, 2014). Agile method is a flexible method (Bhoola & Mallik, 2014; Schwaber, 2009; Mishra, Kumar and Chan, 2010) that delivers rapid qualitative solutions (Bhoola & Mallik, 2014). The market demands fast delivery of software products, and it is crucial that organizations meet these changing requirements in order to survive (Moniruzzaman & Hossain, 2003). Agile method has helped the software industry to sustain customer satisfaction (Bhoola & Mallik, 2014). Traditional plan-driven development methods fail to meet requirements and Agile software development methodologies are turning into a better alternative for software developing firms, as it facilitates the planning process and estimation of the customer needs (Moniruzzaman & Hossain, 2003).

An agile approach advocates lesser documentation (Moniruzzaman & Hossain, 2003), as the focus is on face-to-face communication, interaction, and the sharing of ideas (Guntamukkala, Wen & Tarn, 2006). Better communication leads to an increased transparency and collaboration (Laanti et al, 2010). When working in an agile method you can earlier detect the defects (Laanti et al, 2010), which is seen as a consequence working in shorter iterations.

Agile methods yield several benefits. According to Boehm and Turner (2004) agile development methodologies promise a faster development time, which leads to a reduction of time-to-market (Torrecilla-Salinas et al, 2015). Furthermore, closer collaboration within the organization, higher motivation among team members and increasing work productivity are also benefits of agile working methods (Laanti, Salo & Abrahamsson, 2010). Agile development methodologies assure a higher customer satisfaction (Boehm & Turner, 2004), lower defect rates and a solution to rapidly changing requirements (Moniruzzaman & Hossain, 2003). Furthermore earlier studys show increased quality, reduced waste, better predictability and an increased ability to respond to dynamic market change (Laanti et al, 2011). An agile approach also gives the team members an opportunity to improve and grow, as new ideas and knowledge increases (Melo et al., 2012). Moreover, since companies work more customers centric in the agile approach, the customer satisfaction increase significantly (Ceschi, Sillitti, Succi & Panfilis, 2005).

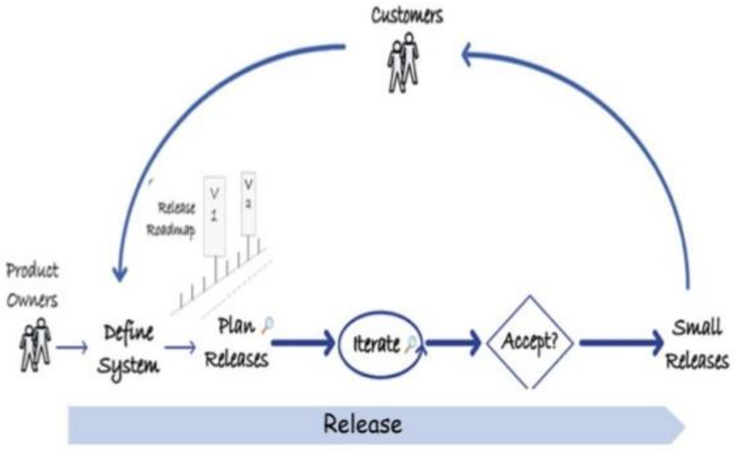

***Figure 1**: Active customer involvement in agile approach*
Source: (Moniruzzaman & Hossain 2003)

As the market in software development is rather unstable during the development process, requirements and the product can not be considered a defined process in agile methods. Having predefined steps might not lead to the outcome you have planned, because the process variance is high in software development. There is a great risk those requirements, technology and even the team composition changes. To meet these conditions it is necessary to have short cycles, as well as frequent and short feedback times. Shorter cycles, which appear in agile methodologies, can help development teams to better handle changing requirements and unpredictable demands that appears during the project (Williams 2007).

However, there is a risk for a lot of resistance when deploying an agile method from a traditional method (Bhoola & Malik, 2014; Dybå & Dingsöyr, 2010; Laanti et al, 2010). This is due to the fact that this often leads to a lot of cultural changes in many organizations. The teams are cross-functional and self-organizing, which claims a more decentralized organization structure (Bhoola & Malik, 2014).

2.3.2 The Scrum aspect of agile method

Scrum is an iterative and incremental development method within agile software development. It means that work is done in shorter iterations, where each iteration is called a sprint. Scrum was initially developed for small co-located teams. The number of team members is few in each development team since scrum advocates a direct and informal communication. Since scrum relies on collaboration and knowledge creation between team members, it does not need any project plans with already completed requirements (Heikkilä et al., 2014).

Scrum is suitable for developing functionality in a volatile environment. It is based on flexibility, adaptability and productivity. In the same manner as in agile software development, earlier studies show that scrum promise to yield increased customer satisfaction, lower defect rates, better time-to-market and a solution to rapidly changing requirements (Boehm & Turner, 2003). During the development process, developers can choose freely which techniques and methods that should be used. In order to maximize the added value to the functionalities, it is crucial that management consistently identify deficiencies or impediments in the development process (Heikkilä et al., 2014).

According to Torrecilla-Salinas et al. (2015), the development process in Scrum consists of five different operational processes:

Standup meeting	A daily meeting in which the team members meet and reconcile that work is progressing satisfactorily. All impediments are identified and discussed in order to fix them.
Product Backlog	Scrum projects start with the creation of a Product Backlog. The product backlog contains of a list of all the features that is demanded during the development process. The features are prioritized in which order they should be completed. The product backlog are a constant changing document, in which featured can be added, deleted, adjusted and re-prioritized as a consequence of changing requirements.
Sprint	A shorter working cycle, which usually run over for 2-4 weeks. It starts and ends with a fixed and expected date, even though work is not completely finished. If that is the case, the remaining work is put back into the backlog.
User story	A piece of functionality that provide the customer with value. It is located in the product backlog. It represents a certain customer need, even though it is not explicit documented. It describes what kind of customer need we are going to satisfy and thereby it works as a reminder for the team member in order to develop functionalities into the right direction. A user story should be characterized by following attributes: independent, valuable, estimable and testable.
Story Point	An estimation of how much time and effort a user story claims. You estimate story points in order to compare different user stories.

***Own table: table 1 -** scrum operational processes*

There are five identifiable roles in Scrum which have different tasks and purposes. Schwaber and Beedle (2002) describe the roles and responsibilities in the Scrum teams as following:

Scrum Master	The Scrum Master is responsible for ensuring that the project is carried through according to the practices, values and rules of Scrum and that it progresses as planned. Scrum master interacts with the project team as well as with the customer and the management during the project
Product Owner	The product owner is officially responsible for the project, managing, controlling and making visible the Product Backlog list. He makes the final decisions of the tasks related to product Backlog, participates in estimating the development effort for Backlog items and turns the issues in the Backlog into features to be developed.
Scrum Team	Scrum Team is the project team that has the authority to decide on the necessary actions and to organize itself in order to achieve the goals of each Sprint. The scrum team is involved, for example, in effort estimation, creating the Sprint Backlog, reviewing the product Backlog list and suggesting impediments that need to be removed from the project.
Customer	Customers participate in the tasks related to product Backlog items for the system being developed or enhanced.
Management	Management is in charge of final decision making, along with the charters, standards and conventions to be followed in the project. Management also participates in the setting of goals and requirements.

Own table: table 2 - *the roles and responsibilities in Scrum teams*

3. Methodology

In this chapter a declaration and motivation for our choice of methodology will be made in order to fulfill the purpose of our study. Furthermore, data collection will be explained and justified and the concepts of validity and reliability will be discussed.

3.1 Field study presentation

Telia.se

This case study is performed at Telia.se, which is a copyrighted website and a part of the cooperation TeliaSonera AB. This site contains both information about the company's services and e-commerce, which makes it complicated since they therefore can not buy ready-made solutions. Telia.se is a portal where employees rearrange functionalities to suit special requirements. In other words, employees work with rebuilding functionality in order to customize the portal as user-friendly as possible for the customers. Telia.se is among one of the first large organizations in Sweden to adopt agile methodologies in their software development, which was made about a year ago. Consequently, there is a lack of empirical evidence in how this affects larger organization’s efficiency in Sweden.

Telia.se is the only department at TeliaSonera that has implemented work with agile methods. Scrum is the established way of working and there are now nine operating scrum-teams. The team members are a mixture of personnel from the IT-department and from the business department. TeliaSonera are an old organization with a strong organizational culture and complex IT system background, which consequently makes it hard to implement a new working method.

3.2 Research design

3.2.1 Abductive approach

A research approach could either be deductive, inductive or abductive. In this thesis we have chosen to use the abductive approach, which has some features of the other two approaches. For the case study methodology, which this study is made with, the abductive approach is the most common research design (Alvesson & Sköldberg, 2011). In different phases of the research process, both inductive and deductive are used (Eriksson & Kovalainen, 2008). The abductive approach means that a particular case is interpreted from a hypothetical overall pattern, which, if it was correct, explains the case in question (Alvesson & Sköldberg, 2011).

During the process when working with the abductive approach, the empirical scope develops gradually and the theory adjusts and refines (Alvesson & Sköldberg, 2011). In this thesis, we start with empirical facts but do not reject theoretical facts when conducting our background and problem discussion.

Since the purpose of this study was to examine whether or not the software development efficiency has increased at telia.se, it was appropriate to use the abductive approach since it combines theory with empirical findings, thus makes it possible to find similarities and differences.

3.2.2 Case study setting

A case study methodology was chosen as a research approach in order to answer the research question. This method gives the opportunity to receive a greater understanding for our specific processes and activities in the organization. According to Eriksson & Kovalainen (2011) a central feature of all case study research is the construction of “the case”, which means that the research question always is related to the understanding and solving of the case.

The case study research is a popular method and one reason for this is its ability to present complex and hard-to-grasp business issues in organizations (Eriksson & Kovalainen, 2011).

Something that is highly valuable in this study since telia.se is an organization within the TeliaSonera AB cooperation. Furthermore, a case study is according to Orlikowski and Baroudi (1991) an appropriate research design when performing research within information systems, as it generates more valid interpretative knowledge compared to other research designs.

3.3 Data collection

3.3.1 Primary Data

Empirical data collected by researchers themselves in a thesis is called primary data (Eriksson & Kovalainen). Saunders, Lewis & Thornhill (2003) describes primary sources as “sources that are the first occurrence of a piece of work” (p. 69). For instance this could be done through observations, surveys or interviews. According to Bryman and Bell (2011) the most used method for qualitative research are probably interviews. Primary qualitative data was collected through numerous semi-structured face-to-face interviews with individuals who possess different positions at TeliaSonera. The purpose of collecting empirical data through interviews was to study the theoretical framework with the results from the interviews.

3.3.2 Secondary Data

Primary data collected through the interviews will be supplemented, analyzed and discussed together with secondary data in the form of documentation and information about telia.se which the organization provide us with. Secondary data means that it has already been collected for another purpose and already exists (Eriksson & Kovalainen, 2011) which should be taken into consideration in the analysis. This data could for instance be books and journals. Information which is important to take into account when processing secondary data is how these data have been collected, where the data comes from, how it is written down and what sources of error that may be relevant (Befring, 1994).

3.4 Interviews

The empirical material was collected through qualitative interviews, which can be done in several ways. This thesis' empirical data collection was made through semi-structured face-to-face interviews. Qualitative, semi-structured face-to-face interviews have both benefits and disadvantages, there are also a few other aspect that is good to highlight.

3.4.1 Semi-Structured Questions

When planning the interviews for this thesis, it was decided to perform these in a semi-structured way. The semi-structured interviews are a mixture of structured and unstructured interviews and it makes it possible to ask follow-up questions and for the interview person to say things that he or she thinks is important for our thesis (Bryman & Bell, 2011). When preparing for these interviews, questions were constructed as a guideline but we kept our mind open for new questions during the interviews and had the possibility to vary the wording and order of questions in each interview (Eriksson & Kovalainen, 2011).

To create the questions for the interviews, we used a theoretical framework to better understand the questions, answers and all of the information that were given during the interviews. After the interviews were conducted, more theoretical information was collected to receive a better understanding for the answers. The interviews were conducted in Swedish; hence this is the native language of all parts involved in the interview. Since this thesis is written in English, the answers are translated to English in the empirical section. According to the qualitative design of our study the primary source of our data are semi-structured interviews. Those allowed us to collect rich data while keeping the flexibility necessary for an explorative study (Stettina & Hörz 2014).

3.4.2 Face-to-face interviews

Referring to Jacobsen (2002), face-to-face interviews can provide more depth in the answers compared to if the interviews are performed over telephone. It is more likely to get an honest answer during face-to-face interviews than when answers are given over the telephone. If the interviews contain many open questions Jacobsen (2002) consequently concludes that it is inappropriate to conduct the interviews by telephone. Furthermore, face-to-face interviews often make the respondent more comfortable and thus may open up more (Jacobsen, 2002) and the ability for the interviewer to observe a respondent's behavior can make it easier to ask follow-up questions (Blumberg, Cooper & Schindler, 2011).

Jacobsen (2002) says that according to Groves and Kahn (1979), there is a significant disadvantage of face-to-face interviews; this is called the 'interviewer effect'. According to Jacobsen (2002) 'the interviewer effect' has lower probability to take place during telephone interviews since the interviewer cannot affect the respondents to the same extent in a telephone interview as during a face-to-face interview. If this effect occurs, the advantages described previously might become disadvantages.

3.4.3 Planning the interview guideline

"Developing research questions into interview questions" (Eriksson & Kovalainen, 2008, p. 79) implies that interview questions should bring out the answers needed in order to answer the research question of the study.

When planning the interview guideline (Appendix 1), a few open questions were formed to make sure the research question could be answered afterwards. The questions were created in a number of categories, and thereafter we made sure that the respondent would be able to talk about each category openly with the questions as guide. The questions for the interviews were formed and based on the individual person's position hold in the company and will therefore not include and take people's gender, ethnicity or similar into account.

3.4.4 Recording and transcription

Recording interviews can be done in a numerous of ways. It can either be done by ongoing notes, notes after the interview or by recording the interview with a recorder machine. According to

Eriksson and Kovalainen (2008), writing ongoing notes may interfere with the interview process, although writing notes after the interview may easily miss out details. These interviews were recorded, both by ongoing notes and by a recording machine, in order to make sure that no details were missed out.

3.5 Respondents

The respondents were chosen by their positions at telia.se. A document with the interview questions were compiled and sent to our contact person, Eriksson, at Telia.se. After some discussion he helped us with the selection of respondents. All respondents work at TeliaSonera's office situated in Gothenburg. A further description of each respondent is attached in Appendix 2. Below follows a compilation of our respondents:

Respondent	**Position**	**Work experience at TeliaSonera**	**Department**	**Date**	**Interview time**
Maria Lund	It solution expert, business analyst	30 years	IT	28/04-2015	50 min
Thomas Alm	Responsible for the IT department	13 years	IT	30/04-2015	50 min
Beatrice Düring	Agile coach	6 years	IT	30/04-2015	55 min
Thomas Trolltoft	Scrum master, development lead	5 years	IT	30/04-2015	45 min
Quintus Lindblom	Product Owner	8 years	Digital Channels	05/05-2015	50 min
Mikael Eriksson	Architect	19 years	IT	05/05-2015	70 min
Martin Hedbäck	Developer, Scrum master	6 years	IT	08/05-2015	30 min
Anders Meyer	Head chief product owner	17 years	Digital Channels	08/05-2015	60 min

Own table: table 3 *– Respondents*

3.6 Data Analysis

After the interviews were done, all information was reduced in order to simplify and get a better structure. When all interviews were transcribed and written down we followed three steps that Jacobsen (2002) has presented in order to analyze qualitative data.

Description

All interviews were described thorough and in detail when written down. We tried to not influence the information given from the respondents (Jacobsen, 2002).

Systematization and Categorization

In the next step, the information was systematized into eight different categories (personnel satisfaction, learning process, customer satisfaction, quality of the functionalities, time-to-market, communication, flexibility and team productivity) to make it easier for us to be able to convey what we found. Thereafter the information were reduced, since all information are to comprehensive (Jacobsen, 2002).

Combination

Finally the data was analyzed by looking at sentences and causes that are similar to bring some order in the data. This was done by comparing the empirical findings against the theoretical findings (Jacobsen, 2002).

3.7 Validity and Reliability

When conducting an empirical investigation, there is always a risk that the result of the study is biased (Jacobsen, 2003). In this study, all the respondents were offered anonymity to avoid misleading answers, however none of the respondents wanted to be anonymous. All of the interviews were tape-recorded which increase the reliability of the transcribed answers. However, the result of the interviews might be biased by telia.se's organizational culture. There is also a risk that the interviews are biased due to the fact that the respondents were not chosen by us, but instead of our contact person at Telia. We are aware of this potentially bias by the respondents, but we do not think that the respondents will have any interest in not telling the truth. Since the interviews where held in Swedish and then translated to English for the study, there is a risk for translation problem. Moreover, a major disadvantage of our approach is the potential bias and subjectivity that result from the inside author's direct involvement in the case.

To gain reliability in the empirical data collection, we have tried to avoid leading questions during the interviews and the empirical data were sent to the respondents to make sure there was no misinterpretation. It was not possible to interview more respondents due to lack of time and resources. There are chances that we would get other answers as well as a greater credibility if more respondents had been interviewed. Furthermore the location was limited to Gothenburg in Sweden and we could therefore not gain information about opinions from employees at telia.se from other locations in Sweden. This could lead to a limitation in the empirical findings since the employees at one office might not have the same opinions as employees at another office. Therefore, the qualitative findings of this study are highly context and case dependent. Most of the respondents belonged to one team, and our data is based on the respondent's perceptions.

Furthermore, it is argued that studying information systems phenomena can be restrictive as the relationships between information technology, people and organizations is highly dependent on each studied context (Orlokowski & Baroudi, 1991). Therefore it is hard to make generalizations from this kind of studies. However, by performing a case study, better valid empirical evidence can be provided (Orlokowski & Baroudi, 1991), as the intention of this research design is to gain deeper understanding of the structure of one phenomenon.

Moreover, the design and use of information technology in organizations depend on social context, time and culture. This study has to some degree ignored the historical context as no exploration about the history and context of the organization has been made. The only assumption made is that telia.se is an operating entity within the corporation TeliaSonera AB, which is large company with complex IT history.

3.8 Ethical position

It is crucial to conduct a research and interviews in a moral and responsible way. It is important to protect the respondents, and the research must be conducted so that they do not experience discomfort, physical harm or loss of privacy (Blumberg, Cooper & Schindler, 2011). To shield the respondents from above mentioned things, the purpose of the study was explained and they were offered anonymity. This motivates the respondents to answer the questions more truthfully (Blumberg, Cooper & Schindler, 2011). Moreover no financial or material reward was offered, as this might lead to bias of the result (Collis & Hussey, 2009). Furthermore, all of the interviewees were clearly informed of the purpose and benefits of the study conducted, to further improve the participation (Blumberg et. al, 2011).

4. Empirical findings

In this chapter, the empirical data from the performed interviews are presented. Since the aim of this survey was to find out if and how agile software development has improved the operational efficiency of telia.se, the empirical material in this study is divided into the eight criterions we have chosen to investigate. Therefore, most of the interview questions were asked in a retrospectively manner. At last, the empirical findings are compiled in a table, which summarizes the respondents' opinions, which facilitate identifying relationships, as well as drawbacks and benefits of agile software development.

4.1 Personnel satisfaction

Overall, all respondents are pleased with the implementation of agile software development and most of them have a positive attitude towards the working method. Lindblom says it is a pleasant and rewarding way of working. Hedbäck agrees with Lindblom, arguing that it is a pleasing working method, which facilitates supporting of your colleagues. Moreover, it facilitates the working process so that team members develop towards common goals. Trolltoft thinks the agile methodology is a natural way of working in software development today. Lund says you feel more involved in the working process in contrast to the waterfall model. The agile way of working is more motivating because you can see the results in a faster pace. Meyer thinks agile methods increase the engagement among team members, which according to him "releases the inner spirit of the employees". However, Meyer argues there is a risk for a contrary effect, which can make the team members feel too controlled. Düring is partly on the same track, when she describes the "agile blind spot". The agile way of working might not suit all people, as each person's' work is very transparent and thereby is there high pressure to deliver. However, this is something they have not experienced yet.

All respondents emphasized that the collaboration between team members is better now. Eriksson and Lund thinks this is due to the closeness between the product owners and the IT-organization today. The product owners must be present and regularly prioritize which leads to better involvement in the development process today compared to earlier when they worked in the waterfall method. Eriksson also argues that the permanent teams with different expertise competencies contribute to better collaboration, as this leads to improved engagement among team members in order to deliver value. Lund agrees with Eriksson. She says the agile teams works all the way from idea to delivery of the functionality, which require collaboration with different kind of competencies. Lund also states that the better collaboration with stakeholders from the business department helps IT to develop functionality which is more strategic right. She thinks the better collaboration with the business department is crucial, as they provide IT with long-term thinking in how telia.se strategically is going to earn money. However, she claims that telia.se needs them even more in their development work, as they contribute with crucial strategic input.

Alm agrees that the collaboration between IT and business has been improved since agile methodology was implemented. He says they exchange demos every week as an example on their improved collaboration. Another improvement is the usage of cross-functional teams today, something that would not have been achievable in the waterfall method. However, Alm argues that the collaboration between the teams is deficient, and therefore they hire agile coaches in order to improve this. Both Alm and Hedbäck mention the expression “scrum of scrum”, as an example on better collaboration. This means the scrum teams have continuous meetings, in which they exchange experiences and knowledge, something that substantially have benefitted the development process.

The agile deployment process at telia.se has been complicated. Meyer elucidate that telia.se still is in the process of fully adopting the agile methodology. He argues that the organization culture still feels resistance of fully placing responsibility down to the teams. He still thinks there are too much bureaucracy and documentation. Eriksson agrees with Meyer, as he initially experienced that it was hard for decision-makers to let go of the control. He feels this works better today though. Alm also argues that it was hard to implement an agile working method. He tells they have tried to implement agile methods earlier, but it did not succeed. His perception is that it has mainly been difficult from a financial perspective, as he experienced difficulties for management to let managers on a lower level take decisions. He also experienced that it was hard for management to accept the fact that they would not exactly know the output, only the outcome of functionality. However, he thinks this has changed since last year and it is more accepted now. He tells that they have established “Lean” as a working principle for the whole corporation. This is related to the agile way of working, which he thinks has improved the implementation process of agile methods this time. But he points out that this problem still concerns telia.se. He believes the whole corporation mentally needs to convert to a more agile thinking. Lindblom agrees with Meyer and Alm, arguing that the implementation process has been challenging. He states that it is important that other units in TeliaSonera must be aware of the agile work on telia.se, which consequently should lead to better understanding.

Meyer says they initialized the implementation process by deciding capacities (resource consumption), and based on these results, they built scrum teams. He argues that it was important to build an adaptive version of agile methodology on telia.se, in order deliver and measure result. He also stated that it is important to have a target vision, so that all team members understood what they were aiming for. Düring thinks agile deployment can be challenging at large organizations, especially if some business activities are outsourced. It is therefore important that the management is interested, engaged and curious in the implementation process. Eriksson argues that the new management of TeliaSonera has improved the implementation process, as they have experienced and understood the implication of agile methodology. According to Eriksson, it is also important to realize the agile methodology does not work in all business units at TeliaSonera. These units can borrow some mindsets from the agile methodology, but not fully implement it.

4.2 Learning process

The learning process has been substantially improved since the implementation of agile software development. Lund experiences that the learning process among team members has improved the longer they work together, which after a while leads to better team efficiency. Earlier in the waterfall method, they had new combined team for each new project. She felt it took time for the project team to learn how to interact in the best possible way, which later was something they never experienced, as they had to move on to the next project. Because of this, she believes that the learning process is improved today due to the permanent teams. Eriksson explains this phenomenon with the FIRO theory, which was one argument to implement agile software development at telia.se. What they had in mind was the need to create efficient teams, as the project teams earlier in the waterfall method constantly changed. When you put together a group of people, it takes a while before they find their roles in that particular social context. They need to identify what kind of people they have to interact with, and it might take a while before they dare to say what they really want to, as they in the beginning are too polite to each other. It is only then the team starts to become efficient. As soon as one person is replaced, the procedure restarts. Eriksson further argues that it is important that you do not break a well-functioning structure. However, the permanent teams in agile methodology has facilitated this problem that earlier existed in waterfall methodology. By using permanent teams and dedicated resources from the business department, you create sustainability when developing software. Eriksson also mention that they struggled with projects, which strived against each other in the waterfall methodology. He believes assigned fields of knowledge with requirements from the business department holds the project's’ outcome better in consensus now. Alm thinks daily standup meetings leads to increased knowledge creation. He mention that they use pair programming, i.e. two programmers works together, as an example of knowledge sharing. He also believes constant evaluations of each sprint contribute to increased learning process.

Many of the respondents argue that an increased responsibility among team members leads to improved creativity in the development process. Trolltoft says he has experienced a better team creativity. He believes this depends on more discussion among team members today, which leads to better knowledge sharing. However, he argues that he does not see a big difference on the individual level. Alm is not sure either whether the creation among the employees has been improved or not. He thinks innovation and creativity always have been present in the development work at telia.se. Eriksson believes creativity to some extent has improved due to less bureaucracy. It is easier to make an impact today, as you do not have to wait for answers from other levels of decision making. Lindblom thinks creativity has increased in the agile work, which he believes is especially visible in the teams. He argues this depends on more viable result, and this gives team members the opportunity to be proud of their results.

4.3 Customer satisfaction

The customer satisfaction has been significantly improved since the adoption of the agile methodology. This is measured by the index NKI ("Nöjd Kund Index - "satisfied customer index"), which shows a distinct trend upwards. Meyer says the NKI declined when telia.se worked with the waterfall method, but this trend reversed as soon as they implemented agile software development. Except for an increased NKI, Meyer argues that you obviously notice an improved customer satisfaction. He believes this is partly due to the fact that it is easier to remove obstacles for customers, as it easier to correct bad solutions today. This was not the case in the waterfall method, as it consisted of slow-moving projects. Lindblom says the telecom industry still suffers from bad reputation, since the customers generally are quite dissatisfied with the telecom industry. Telia has a quite good reputation, therefore is it important they maintain this position. Agile software development facilitates this work, as it is natural to perpetually validate against stakeholders, where the customer is the main stakeholder. He argues that there still is a risk you might release functionality that the customers do not like. However, Lindblom claims that they have better tools to fix these problems nowadays. You discover the unpleasant functionality earlier, and you can repair it in a faster pace thanks to the shorter iterations in the agile work. Alm believes the better customer satisfaction depend on improved time-to-market speed. Despite this, Eriksson argues that there is still room for improvements. The team members must identify functionalities, which leads to more sales and improved value-added. Hedbäck also points out that he sometimes demands more iterations in order to validate against customers more frequently. He believes this would increase the quality of the functionalities.

4.4 Quality of the functionality

Some of the respondents believe the quality of the functionality is improved in the agile work. Düring mentions that she can see a slightly improvement on the quality. Hedbäck agrees, saying you can see an up going trend, but this is still a work in progress. However, he thinks working in shorter iterations is the main success factor to an increased quality. As a consequence, they maintain better focus on what is topical right now. Lindblom argues that greater responsibility taking in the agile method leads to better quality. He claims he can see an improvement on the quality of the functionality solutions at telia.se. And in the case of poor functionality, they can fix it more easily now. He and Trolltoft argue that you discover defects earlier today, also due to the shorter iterations.

Though, Lund argues functionality sometimes is too short-oriented in agile method. On the other hand, she argues that this might be the right strategy when operating in a world of constant change. But she believes the development-process sometimes need some form of pre-study in order to deliver long-term solutions. Moreover, Lund claims that the functionality have better quality today due to the better collaboration with stakeholders. They are more involved during the development process, which facilitate the "common-thread" work. Lindblom and Alm also mention there is a risk of too short-oriented functionality, in other words they have too short life-length. However, they have implemented guidelines regarding architecture and appearance in order to avoid this risk. On the other hand, Alm argues that this was riskier in the waterfall method as it is based on delivering within time and budget. It is easy for the team members to take shortcuts when the team is exposed to pressure, which leads to decreased quality on the functionality. He believes the team is not exposed to the same level of pressure under the agile work. On the contrary, Trolltoft argues that there is a pressure to deliver fast-pace functionality today, something that decreases the quality. Both Alm and Trolltoft claim they do not see any noticeable improvement of the quality of the solutions though. Alm believes that he sees a large development potential of the solutions. He suggests that the teams should be responsible for a more distinct area, which should lead to a quality-enhancing effect. Trolltoft states you focus on "doing the right things" rather than "doing the things right", as the latter one does not contribute to any value if it is not up-to-date.

4.5 Time-to-market

Almost all respondents agree that time-to-market has improved since the adoption of agile software development. However, Trolltoft does not fully agree, as he thinks they delivered quick functionality earlier as well in occurrence of a stressful situation. Lindblom thinks TTM (time-to-market) has been improved, and argues this is due to shorter iterations. Instead of delivering the whole product at once, they build little by little. Thus, they can quickly pick up market changes. Eriksson says it is due to the efficient teams that time-to-market has been improved. He says lead times were longer in the waterfall method, as they had much more bureaucracy. The decision-making process needed to go through several stages before they could continue the development process, which lead to inefficiency. Agile software development claims a more decentralized organization structure, which moves the decision-making down to operation level. Consequently, the decision-making processes are shorter which facilitates a faster TTM. Furthermore, Eriksson argues that the learning process among product owners is crucial in order to make the right prioritization. When doing so, the IT department will receive rapid response, which shortens TTM. However, Lund says there is still room for improvements. She argues they must dare to release developed functionality more frequently.

4.6 Communication

It is clear that communication has been improved since the adoption of agile software development. Meyer argues that the improved communication partly depends on the more decentralized organization structure today. Lund believes especially the communication between the business and IT department has been improved. Lund experiences that there is a conflict of interest between IT and business. The IT department might prioritize functionality with better quality, which has a more time consuming development process. On the contrary, the business department demands functionality which have a shorter time-to-market. However, she argues that this conflict have been reduced since the adoption of agile software development. She believes this depends on better understanding among team members which is a consequence of better communication. She believes the communication between team members are better now compared to the waterfall method, arguing agile method is more or less a model built on daily communication. However, she argues that communication is time consuming. She thinks they spend more time on meetings today than before in waterfall method. She thinks it is important that each meeting generate value, so they do not waste resources on non-value bringing processes.

Hedbäck thinks the agile methodology leads to a more open culture, in which it is friendlier to talk to each other. Hedbäck also experiences the conflict of interest between business and the IT department, and finds this as one of the main challenges at telia.se. He believes agile partly reduced this conflict, as it leads to better communication and understanding from departments. He mentions the daily stand ups as an example on improved communication within the teams. This was not the case in waterfall, as the projects often turned out to be hermetic. Eriksson argues that the increased communication is due to people's interdependency in the agile methodology. Furthermore, he argues that greater responsibility leads to an improved communication, as this is crucial for each person's deliverable. Lindblom argues more frequent assessments in the agile work today are one reason to the improved communication. Trolltoft agrees, saying there is an excellent communication today. However, he argues that this was satisfying in the waterfall method as well, and points out the will to communication depend on what kind of person you are. However, he believes agile methodology gives better tools for improved communication.

4.7 Flexibility

Flexibility has increased since the agile deployment at telia.se. Eriksson argues that you always can re-prioritize if you notice that market has changed. Furthermore, they do not have the same chunks of requirements today as you had in waterfall. Then it was crucial that requirements were well understood due to the long lead-times. The delivered functionality was often outmoded, and did not work against the customer. Then it is a big advantage to work agile, since they continuously can evaluate the market. They work in shorter iterations, which make the solutions more up-to-date. When they worked in waterfall method, they often released functionality that was passé. Furthermore, Eriksson argues that the flexibility has increased due to the fact that they discover defects earlier today. The respondents also mention that they measure customer satisfaction, which give them groundwork to re-prioritize and validate against customer. This was not possible in the waterfall method, as they had to deliver requirements that were set-up initially before starting the project. Furthermore, Eriksson argues that the continuously capacities and the opportunity to re-prioritize facilitates flexibility. Regarding this, the flexibility has increased significantly.

4.8 Team productivity

It is hard to measure whether team productivity has been improved or not. Though, what they have seen so far is that it has slightly increased. This is partly due to the fact that they worked in projects earlier. However, Düring says that you should see an improved productivity, better time-to-market, better quality and cumulative lesser costs in order to see a better team performance. She argues that telia.se still has not achieved this yet though. This is a work in progress, and you should see this effect within 6 to 24 months. However, Düring argues that she feels a great curiosity among the team members, which should improve the team performance in the future. Meyer argues that it is hard to measure whether the productivity has changed as a consequence of the agile development. They believe they earn more money now in contrast to what they invested. However, since they did not follow-up projects to the same extent in waterfall method, they are not really sure whether this has changed. Hedbäck also says that it is hard to measure whether the team productivity has changed as they have increased their working capacity. They have hired many junior consultants lately that are not familiar with telia.se. He still believes they have increased their amount of head counts though. He also states that they are in the middle of their deployment process, which affects their productivity. However, Eriksson says that the permanent teams should lead to improved team productivity due to the FIRO model. He argues that the longer a group of people have worked together, the more productive do you get. Even though a team member is replaced in the agile methodology, you do not start from scratch regarding productivity as you might do in the waterfall method. In the waterfall method, each new project consisted of a new composition of people, which accordingly decreased productivity.

4.9 Summarized table

Criterion on software development efficiency	**How was this affected after the adoption of agile software development?**
Personnel satisfaction	Personnel satisfaction has increased. Employees argue that it is a pleasant working method that facilitates motivation, collaboration and engagement. Though, collaboration between teams is still deficient. However, it is argued that there is a risk that you feel to controlled as consequence of transparency. This was called the agile blind spot. The agile deployment process has been complicated though. One of the biggest perceived problems was that the organizational culture still feels resistance to fully placing responsibility down the teams.
Learning process	The learning process has improved substantially since the implementation of agile software development. The permanent teams are seen as the main explanation to an improved learning process, which is explained by the FIRO theory. Work in shorter iterations is also seen as a contribution to an increased learning process.
Customer satisfaction	The customer satisfaction has been significantly improved since the adoption of an agile approach. One of the main explanations is continuously validation against customers. Key performance index as NKI has increased. There is still room for improvements though.
Quality of the functionality	It is unclear whether how much this has changed due to different perceptions. Some respondents can see a slightly improvement. However, some of the respondents think that there is a risk for too short-oriented functionalities.
Time-to-market	Time-to-market has been improved. It is argued that this depends on the more efficient teams and the more decentralized organizational structure.
Communication	Communication has been improved. This is seen as a result on the more decentralized organizational structure and people's interdependency to each other. There is a perceived conflict of interest between the business and IT department though. However, it is argued that this has been reduced thank to the agile approach.
Flexibility	Flexibility has increased. This is seen as a consequence of continuous capacities and the opportunity to re-prioritize.
Team productivity	This was hard to measure since they did not have much measureable result from the work in waterfall method. The teams have not lived long enough to be able to measure how productivity is affected yet. However, they have experienced a slightly increased productivity.

***Own table: table 4** – summarizing of the empirical data*

5. Analysis

In this chapter will the empirical findings received during the interviews utilize to answer the research question of this study. Each of the criterions on software development efficiency will be discussed, as the purpose of the study was to gain better insights in how an agile approach improves software development efficiency. Thus, the discussion given in this chapter will be done from an efficiency perspective. Finally a discussion will be given in order to deliberate arisen thoughts when analyzing the empirical data.

5.1 Personnel satisfaction

This study shows that personnel satisfaction has increased with an agile work. Majority of the respondents believe that agile method is a pleasant and rewarding way of working. Among others, outcomes as better collaboration and engagement among the employees were identified since the agile deployment. This was mainly seen as consequences of the permanent teams. These findings are more or less consistent with earlier studies from example Laanti et al. (2011). Though, the difference between our theoretical research and earlier studies is that collaboration between teams was perceived as deficient. Moreover, our empirical findings show that the personnel feels more involved in today's work, which accordingly leads to better team performances as the motivation increases. This is also in line with earlier studies (Laanti et al., 2011).

However, another interesting finding from this study was the risk for a contrary effect from what was mentioned above. It was argued as this way of working might not suit all employees, due to the transparency and thereby pressure to deliver. This was called the agile blind spot, and if this was the case, it would definitely discourage an efficient work as this makes the personnel feel uncomfortable. Agile is more or less a working method whose success heavily relies on the people that practice it. Therefore it is crucial with satisfied personnel in order to deliver the best possible value to the company. Furthermore, this study showed that the biggest perceived problem regarding personnel satisfaction in the agile work was and still is the deployment process. This is line with earlier studies, which showed that this problem mainly depends on the change of organizational culture, which makes it hard to implement (Bhoola & Malik, 2014; Dybå & Dingsöyr, 2010; Laanti et al, 2010). This study also shows that it has been complicated in terms of antagonism from organizational culture. One interesting finding was that this was mainly from a financial perspective, as management does not exactly know the output from an agile work, only the intended outcome. It was argued that it is important that the whole corporation must adapt to a more agile thinking, in order to not feel resistance of letting teams take the fully responsibility for delivering. However, this study found that this situation has been improved the longer they have worked in the agile approach. As a consequence, this might only be a problem in the beginning of the deployment process.

According to what have been mentioned above, it is suggested that an agile approach facilitates a more efficient work as the personnel satisfaction has increased (Chow & Cao, 2008; Highsmith, 2004). Though, there are risks for pitfalls, which companies must bear in mind when implementing the agile approach

5.2 Learning process

Our study revealed that the learning process has been improved since the adoption of the agile approach. This is consistent with earlier studies, which states that an agile approach gives team members an opportunity to grow and improve, as new ideas and knowledge increases (Melo et al., 2009). Our study showed that the improved learnings process mainly was due to the work in permanent teams. However, a new interesting finding was that this was explained by the FIRO theory, which says it takes time before a group of people get efficient due to that they are not fully comfortable with each other in the beginning. In other words, an agile work facilitates this process. Another finding from this study is that working in shorter iterations leads to better learning process. This is since they continuously can evaluate each sprint, which creates a learning process. Furthermore, it was not clear stated whether creativity has increased since the agile approach was implemented, as the answers varied from different respondents.

Based on what has been mentioned above, it is believed that efficiency has been improved in agile work, as this partly can be derived from the improved learning process.

5.3 Customer satisfaction

This study found that customer satisfaction has increased since agile software development was implemented. This is in agreement with earlier studies, which also promise higher customer satisfaction (Boehm & Turner, 2004; Ceshi et al., 2005). This has been measured by the ratio NKI (Nöjd Kund Index), which has a clear up going trend since the agile deployment. This outcome is a repercussion of the opportunity to customize better functionalities to the customers, which is a consequence of better execution. One finding was that agile facilitate better validation against the customer due to the work in shorter iterations, which is believed to help discovering and eliminate obstacles for customers. According to March (1991), better execution in the development process is a sign on improved efficiency. Though, our study shows that there is room for improvements regarding customer satisfaction, which however an agile work could facilitate. Especially more frequent iterations were requested.

Due to the literature and empirical findings, we argue that an agile approach has considerable positive effect on efficiency regarding customer satisfaction (Ramirez & Nembhard, 2009; Melo et al., 2009).

5.4 Quality on the functionalities

It is not clear whether if and how much the quality of the solutions has changed since the implementation of the agile approach, as the material varied from different respondents. This is a big difference from earlier studies, which research promised high quality solutions (Moniruzzaman & Hossain, 2003; Laanti et al, 2011). On the one hand, some believed that there have been some quality improvements as a result from developing functionalities in shorter iterations, which leads to earlier detections of bugs and improved tools to fix them. In such case, this is line with earlier studies (Moniruzzaman & Hossain, 2003; Laanti et al, 2011). On the other hand, it was argued that increased quality was a work in progress. It is important to have in mind that telia.se recently started working in an agile approach. Improved quality of the functionalities might be not easy to identify yet.

However, one interesting finding from our study was that some of respondents argued for too short-oriented functionalities. If this is the case, this will decrease efficiency, as this do not create continuity in design, which according to Clutterbuck, Rowlands and Seamons (2009) is crucial in order to be efficient in the development processes. Another finding was that that high pressure on delivering fast pace solutions decreased the quality as well. Despite this, another finding is that the work at telia.se has become more strategic “right” after the implementation of agile methods, in terms of that stakeholders are more involved in the development process. The outcome of this could be fewer misunderstandings, which according to Trendowicz and Münch (2009) is something that indicates better efficiency.

Based on our study regarding quality of the solutions, we are not sure whether we can state that the improvements have been sufficient enough in order to contribute to a more efficient work at telia.se.

5.5 Time to market

Findings from this study show that time-to-market has increased, which is in consistent with earlier research (Moniruzzaman & Hossain, 2003; Boehm & Turner, 2004; Torrecilla-Salinas et al., 2015; Boehm & Turner, 2003). This is mainly seen as a consequence from working in shorter iterations, which leads to that functionalities are delivered little by little instead of one large products at once. Another important finding is that a shorter time-to-market is enhanced by the more decentralized organizational structure. This leads to less documentation and bureaucracy, which facilitate a more efficient work due to decreased lead times in the decision making process. With this in mind, we can state that an agile approach, considering the increased time-to-market, has lead to an increased software development efficiency (Melo et al., 2009; March, 1991).

5.6 Communication

The results of this study show that communication has been improved at telia.se with an agile approach, which is in line with earlier research (Laanti et al, 2010; Heikkilä et al, 2014). This is mainly seen a consequence from the more decentralized structure. One interesting finding is that an agile approach especially facilitated communication between two departments, which consequently reduced an interest of conflict between these. Another interesting finding was that our study showed that people's interdependency to each other in the agile approach ease the communication between employees. Better communication in the agile approach is also seen as a consequence from a more open culture with more frequent assessments, which facilitates communication.

According to Buck et al (2009), the outcome of better communication could be derived to better quality of the interaction, which according to them, should lead to better efficiency. With earlier studies and our research, we can argue that an increased communication leads to a more efficient work.

5.7 Flexibility

Our study states that the flexibility has increased since the adoption of agile methods. This is also in line with earlier research (Bhoola & Mallik, 2014; Schwaber, 2009; Mishra, Kumar and Chan, 2010). The option re-prioritize and selection of which functionalities that should be produced and released, contribute to better flexibility. Findings from our study show that when working in the waterfall method, functionality was often passé when released due to the large projects. This is also consistent with earlier research (Thomas 2001; Jarzombek 1999; Johnson 2002; Cohen et al. 2001). However, this problem is reduced in the agile approach, since the teams are more flexible and continuously can evaluate the market. With this in mind, we argue that efficiency has increased in an agile approach due to the increased flexibility (Melo et al.,2012; Clutterbuck, Rowlands and Seamons, 2009).

5.8 Team productivity

Our study could not clearly define whether the team productivity has changed, as there has been a lack of documentation of team productivity in the waterfall method. Our study shows a slightly perceived increased productivity, in which the permanent teams are seen as one explanation. Even though that telia.se still is in the deployment process, this outcome differs a lot from earlier studies, which promised higher productivity (Laanti et al., 2010). However, our study shows that the perception is an increased outcome in contrast to what they invest today. This is believed to be a consequence of the permanent teams. Yet, our study reveals that they believe that their productivity will increase in the future. With our empirical evidence in mind, we can not do any clear statement on that productivity has increased due to an agile approach.

5.9 Discussion

When conducting the interviews, it was clear that some benefits and challenges of agile software reappeared. Our study showed a great satisfaction with the new way of developing IT functionality. Some respondents could barely not see any disadvantages using this working method. However, it is important to set this in the context that the old way of developing functionality in the waterfall method was very outmoded, and therefore this new way of working was very pleasing for them. As mentioned earlier, there have not been made any exploration of the historical context of telia.se's earlier situation. As Orlokowski (1991) argues, the historical context can have considerable large affection on satisfaction levels and the other empirical findings as well. As we have chosen a case study as research approach, it is the respondents perception that creates the result of this study. Therefore it is important that the reader have this in mind, even though we believe that this study will contribute with a lot of understanding when implementing an agile method and to which extent this could affect the software development efficiency.

When performing this study, we could identify many substantial risks and perceived benefits regarding efficiency in agile software development. The biggest perceived problem was the agile deployment process, which also was in line with earlier studies (Bhoola & Malik, 2014; Dybå &

Dingsöyr, 2010; Laanti et al, 2010). In this study, it was antagonism from the old organizational culture that was the mainly perceived problem, as it felt resistance in letting down the fully responsibility to the team members. This was mainly due to the uncertain output from the development process which mainly created hesitation from management. This is however, something you cannot avoid in the agile methodology. Though, we believe this is comprehensible since it is challenging to perform a business plan if the output from the business is uncertain. Accordingly, we believe that management role in an agile work is crucial in order to make it work as efficient as possible. They must dare to give responsibility to operation levels, in order to make the development process as smooth as possible. However they must also guide personnel with clear goals and vision, in order to avoid contradictory work. Thus, we argue that uncertain output is one of the main challenges in the agile method and something that must be more researched in order to find a more efficient solution.

As mentioned above, we believe there is a substantial risk for contradictionary work in an agile development. Even though our study showed that an agile approach to the development process generate a more strategic right work, we believe this can be the short term outcome. Although work is more efficient when the letting down the decision making on operational levels as it leads to shorter lead-times, more people have the ability to make decisions and consequently make the wrong strategic choice due to the difficulty of seeing the overall thinking in the organization from an operational level. As our study showed deficient communication between teams in combination with uncertain output from each team, we argue that there is a great risk of contra dictionary work. Once again, we believe management role is very important as they must set guidelines and clear directive on developed functionality in order to be in consensus with other teams developed functionality and systems. In the case of contradictionary work, it would significantly decrease the long-term efficiency of the firm as it would prevent

To draw a conclusion from the discussion given above, we believe it is the consistent thread in the developed functionality that is one of the biggest challenges, which consequently sets high pressure on management to provide clear visions and guidelines on developed functionality. As almost all businesses today must perform some kind of IT business in order to sustain competitive against other firms with pressure to be agile in order to meet customer demands, this complex of problems will concern many firms in the near future.

6. Conclusions and Further Research

In this chapter the conclusions from our study will be drawn. Furthermore, suggestions on further research will be given in consideration on perceived challenges regarding the efficiency aspect in the adoption process.

6.1 Conclusion

The purpose of this study was to gain better insight into agile software development and if this implementation at telia.se has led to better software development efficiency. By studying theoretical facts and gathering empirical data from interviews with employees at telia.se, we came up with the following conclusions:

Overall, empirical findings of this study shows that an agile approach contributes with better software development efficiency. Most of our investigated criterions have been improved in an agile work, which indicates an increased contribution of efficiency in an agile development process. However, the criterion, which was supposed to measure team productivity, had insufficient material in order to make a clear statement. Similar with the criterion which was supposed to estimate quality of the functionalities, since the requested material varied from different respondents, we could not draw any clear statement from this field of study. However, we have identified some distinctive characters from our study.

Findings from our study revealed that permanent teams are seen as one of the largest contributions to improved efficiency when working in agile software development. The permanent teams were mentioned several times during the empirical collection as evidence on improved efficiency, since they contributed to better collaboration, knowledge creation, and team productivity. According to our theoretical research, this was not discernible. Furthermore, the shorter iterations was also mentioned several times as explanation to better efficiency. This was however, more or less in in consistent with earlier studies. Another interesting finding was that our study did not reveal the same results as earlier studies have regarding quality of the functionalities. In our study, higher quality of the functionalities would be a highly uncertain statement. In earlier studies, this is a quite common result. However, it must be taking in consideration that telia.se still is in the adoption phase of agile software development, and therefore the result could be different if the time frame of the study was extended.

One of the biggest perceived problems of agile software development was the agile deployment process. Antagonism was mainly perceived from the organizational culture, which felt resistance in letting down the fully responsibility to the team members. Moreover, our study showed that uncertainties regarding the output were a significant problem. As we argued in the discussion given above, it is believed that management role is crucial in order to solve this kind of problems.

Furthermore, our study revealed a potential risk for an agile blind spot, which states the agility's high requirements on skilled personnel with special features, which only few people possesses. When working in a large organization, this can be hard to manage properly, as these people are coveted in all organizations, which lead to not everyone in an agile work, have these requested features. Thus, it is harder to operate in an efficient manner. In other words, it is questionable to which extent an agile methodology suits all kinds of people.

6.2 Theoretical and empirical contribution

This study has contributed to see how an agile approach works in a larger organization by comparing existing theoretical framework to empirical findings from the case study at telia.se in order to identify similarities, as well as disparities, between the two. The practical contribution of this study enhances the understanding of what kind of benefits and problems occur when adopting an agile software development within a large organization, and to which extent this affects the software development efficiency of the firm. Findings of this study suggest that software efficiency increases when adopting agile approach to the development process, however there are some risk against efficiency which firms must bear in mind when adopting an agile approach. Thus, findings of this study can be valuable for firms who consider adopting agile software development as it provide them with tips and new ideas. The theoretical contribution of this study encourages a discussion regarding software efficiency and contingent challenges to an efficient work when working agile. As there is a lack of empirical evidence whether how software development work in practice, this study fill this gap as it provide literature with deeper understanding within this particular context. Thus no generalizations from this study can be made, however the result still contributes with important aspects to have in mind when adopting an agile approach to the software development.

6.3 Further research

We would like to suggest further researches in how management can plan the operational work in a more efficient manner in agile software development. In today's situation, it is common that management sets long strategic visions as the operational work is hard to plan properly without knowing the output from work. This risks the common thread when working on middle management level.

References

Abrahamsson, P., (2000). *Measuring the success of software process improvement: the dimensions*. Proceedings of European Software Process Improvement Conference.

Abrahamsson, P., Warsta, J., Siponen, M. T., Ronkainen, J., (2003). New Directions on Agile Methods: A Comparative Analysis. Technical Research Centre of Finland. VTT Electronics.

Abrahamsson, P., Salo, O. and Ronkainen, J., (2002), Agile Software Development Methods – Review and Analysis, VTT Electronics, No. 478, Oulu. Improving software organizations 379

Alvesson, M., Sköldberg, K., (2008). *Tolkning och Reflektion - Vetenskapsfilosofi och Kvalitativ metod*, 2nd edn, Studentlitteratur, Lund

Ambler, S.W., (2008). *Scaling Scrum: Meeting Real-World Development Needs.* Dr. Dobb's Journal.

Arbnor, I., Bjerke, B., (1994). *Företagsekonomisk metodlära.* Lund: Studentlitteratur.

Atkinson, R., (1999). *Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria.* International Journal of Project Management Vol. 17, No. 6, pp. 337-342.

Austin, R., Devin, L., (2003). Artful Making: *What Managers Need to Know About How Artists Work*, Boston: Financial Times Prentice Hall.

Barlow, J., Giboney, S. G., Keith, M. J., Wilson, D. W., Schuetlzer, R. M., Lowry, P.B., Vance, A., (2011). *Overview and Guidance on Agile development in large organizations.*

Befring, E., (1994). *Forskningsmetodik och statistik*. Lund: Studentlitteratur AB.

Boehm, B., (2002). *Get ready for Agile methods, with Care!*

Boehm, B., Turner, R., (2003). *Using risk to balance agile and plan-driven methods.* IEEE Software. Vol. 36, Issue 6, pp. 57-66

Boehm, W., (1976) *Software engineering.* IEEE Transactions on Computers C25(12).

Boehm, B.,Turner, R., (2004*). Balancing Agility and Discipline: Evaluating and Integrating Agile and Plan-driven methods.*

Bhoola, V., Mallik, D., (2014). *Determinants of Agile Practices: A Gini index approach.*

Bhattacherjee, A., (2001). *Understanding information systems continuance: an expectation-confirmation model*. MIS Quarterly, 25, 3, 351–370.

Bjarnson, B., Wnuk, K., Regell, B., (2012). Are you biting off more than you can chew? A case study on causes and effects of overscoping in large-scale software engineering

Bloch, M., Blumberg, S., Laartz, J., (2012). *Delivering large-scale IT projects on time, on budget, and value.* McKinsey Quarterly.

Bloor, M., Wood, F., (2006). *Keywords in Qualitative Methods, A vocabulary of research concepts*. SAGE Publications Ltd

Blumberg, B., Cooper, D.R., Schindler, P. S., (2011). *Business Research Methods*. Berkshire: McGraw Hill Education.

Bryman, A., Bell, E., (2005). *Företagsekonomiska forskningsmetoder.*

Börjesson, A., and Mathiassen, L., (2005*). Improving software organizations: agility challenges and implications*. Information and Technology & People, Vol. 18 Iss 4 pp. 359-382

Ceschi, M., Sillitti, A., Succi, G., Panfilis, S.D., (2005). *Project management in plan-based and agile companies*. IEEE Softw 22(3):21–27.

Chow, T., Cao, B,. (2008) A survey study of critical success factors in agile software projects, Journal of Systems and Software 81 (6) 961–971.

Clutterbuck, P., Rowlands, T. and Seamons, O. (2009). *A Case Study of SME Web Application Development Effectiveness via Agile Methods.* The Electronic Journal Information Systems Evaluation Volume 12 Issue 1.

Cockburn, A. (2001). Agile Software Development, Boston: Addison-Wesley.

Cockburn, A., (2001). Agile Software Development, Boston: Addison-Wesley.

Coelli, T. J., Rao, P. D.S., O'Donnell, C.J., Battese, G.E., (2005). *An introduction to efficiency and productivity analysis*. Second edition.

Cohen, D., Larson, G., Ware, B., (2001). *Improving software investments through requirements validation*. Proceedings of the 26th annual NASA Goddard software engineering workshop. IEEE Computer Society, Washington, p 106

Collis, J., Hussey, R., (2009). *Business research - A Practiful Guide for Undergraduate and Postgraduate Students.* New York: Palgrave MacMillian.

Conboy, K., and Fitzgerald, B., (2004). *Toward a Conceptual Framework of Agile Methods*, in Proceedings of the 2004 ACM Workshop on Interdisciplinary Software Engineering Research, Newport Beach, CA, November 5, 2004, pp. 37-44.

Conboy, K., (2009). Agility from first principles: Reconstructing the Concept of Agility in Information Sytems Development. Vol. 20, No. 3, September 2009, pp. 329–354

Danvea, M., Veen, E., Amrit, C., Ghaisas, S., Sikkel, K., Kumar, R., Ajmeri, N., Ramteerthkar, U., Wieringa, R., (2012). *Agile requirements prioritization in large-scale outsourced system projects: An empirical study.*

Dove, R., (2001). *Response Ability – the Language, Structure, and Culture of the Agile Enterprise*. Wiley, New York, NY.

Drury-Grogan, M. L., (2013). *Performance on agile teams: Relating iteration objectives and critical decisions to project management success factors.*

Dubey, A., Jain, A., Mantri, A., (2015). *Comparative study: waterfall v/s agile model.*

Dybå, T., Dingsøyr, T., (2008). *Empirical studies of agile software development: a systematic review.* Inf. Softw. Technol. 50, 833–859.

Eriksson, P., Kovalainen, A., (2008). *Qualitative Methods in Business Research*, Sage Publications Ltd

Esaisson, P., Giljam, M., Oscarsson, H., Wängnerud, L., (2012). *Metodpraktikan, konsten att studera samhälle, individ och marknad*. Norstedts Juridik AB

Frökjaer, E., Hertzum, M., Hornbaek, K., (2000). *Measuring Usability: Are Effectiveness, Efficiency, and Satisfaction Really Correlated?*

Garg, A., (2009*). Agile Software Development*.

Golden, W., P. Powell. (2000). *Towards a definition of flexibility: In search of the holy grail?* Omega 28 373–384.

Goldman, S., Nagel, R., Preiss K., (1995). *Agile Competitors and Virtual Organisations: Strategies for Enriching the Customer.*

Guntamukkala, V., Wen, H. J., Tarn, J. M. (2006). *An empirical study of selecting software development life cycle models.* Department of Accounting and MIS, Harrison College of Business.

Heikkilä, V.T., Paasivaara, M., Rautiainen, K., Lassenius, C., Toivola, T., Järvinen, J., (2014). *Operational release planning in large-scale Scrum with multiple stakeholders – A longitudinal case study at F-Secure Corporation.*

Hellenius, R., (1991). *Förstå och bättre veta : om hermeneutiken i samhällsvetenskaperna.* Stockholm: Carlsson..

Highsmith, J., (2004). *Agile Project Management – Creating Innovative Products*. Pearson Education, Boston.

Hirsch, M., (2005). *Moving from a plan driven culture to agile development.* Proceedings of the 27th international conference on software engineering

Hong, W., Thong, J., Chasalow, L., and Dhillon, G., (2011*). User acceptance of Agile Information Systems: A Model and Empirical Test*. Vol. 28, No. 1, pp. 235–272.

Jacobsen, D., I., (2002). *Vad, hur och varför? - Om metodval i företasekonomi och andra samhällsvetenskapliga ämnen*. Författaren och Studentlitteratur.

Jalote, P., Palit, A., Kurien, P., (2004). *The Timeboxing Process Model for Iterative Software Development.*

Jarzombek. J., (1999). *The 5th annual jaws s3 proceedings*.

Jin-Hai, L., A. Anderson., R. Harrison., (2003. *The evolution of agile manufacturing. Bus. Process Management J*. 9 170–189.

Johnson. J., (2002). *Keynote speech: build only the features you need.* Proceedings of the 4th international conference on extreme programming and agile processes in software engineering.

Jones, C., (1995). *Patterns of software systems: failure and success*. International Thomson Computer Press, Boston

Koch, C., (2006). *The Truth about SOA. CIO*

Krebs, J., (2008). Agile Portfolio Management. Microsoft Press.

Laanti, M., Salo, O., Abrahamsson, P., (2010) *Agile methods rapidly replacing traditional methods at Nokia: A survey of opinions on agile transformation*.

Lee, G., Xia, W. (2010). *Toward agile: An integrated analysis of quantitative and qualitative field data on software development agility*. MIS Quarterly Vol. 34 No. 1.

Leffingwell, D., (2007*). Scaling Software Agility: Best Practices for Large Enterprises*. Addison-Wesley Professional.

Leffingwell, D., (2010). Agile Software Requirements*: Lean Requirements Practices for Teams, Programs, and the Enterprise*. Addison-Wesley Professional.

Maxwell, K. D., Forselius, P., (2000). *Benchmarking software development productivity*. IEEE Software.

Melo, C., Cruzes, D. S., Kon, F., Conradi, R., (2012). *Interpretative case studies on agile team productivity and management.*

Miller, G. G., (2001). *The Characteristics of Agile Software Processes.* The 39th International Conference of Object-Oriented Language and Systems (TOOLS 39), Santa Barbara, CA.

Mishra, N., Kumar, V., Chan, F.T.S., (2010*). A multi-agent framework for agile outsourced supply chains. Exeter Business School, Exeter.*

Moniruzzaman, A B M., Dr Hossain, S. (2003). *Comparative Study on Agile software development methodologies.*

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

Orlikowski, J., (1991), Integrated information enviroment or matrix of control? The contradictory implications of information technology. Massachusetts Institute of Technology

Petersen. K., Wohlin, C., (2010). *The effect of moving from a plan-driven to an incremental software development approach with agile practices*

Ramirez. Y.W,.Nembhard,. D.A, Measuring knowledge worker productivity:a taxonomy, Journal of Intellectual Capital 5 (4) (2004) 602–628.

Raschke, R. L., (2010). *Process-based view of agility: The value contribution of IT and the effects on process outcomes.*

Rising, L., Janoff N.S., (2000). *The Scrum Software Development Process for Small Teams,* IEEE Software.

Royce, W.W., (1970) *Managing the development of large software systems*. Proceedings of IEEE WESCON, Los Alamitos, CA, IEEE Computer Society Press.

Saunders, M., Lewis, P., Thornhill, A. ,(2009). *Research methods for business students.* Pearson Education.

Schneiderman, A. M., (1996). *Metrics for the order fulfillment process* (Part 1). J Cost Manage.

Schwaber, K., Beedle, M., (2002). *Agile Software Development With Scrum.* Upper Saddle River, NJ, Prentice-Hall.

Schwaber, K., (2009). Agile Project Management with Scrum. Microsoft Press, Washington.

Selic, B. (2009). *Agile Documentation, Anyone?* IEEE Software (26)6, pp. 11–12.

Stettina, C.J., Hörz, J. (2014). *Agile portfolio management: An empirical perspective on the practice in use*. International Journal of Project Management 33.

Svensson, H., (2005). *Developing Support for Agile and Plan-Driven Methods*. Royal Institute of Technology Department of Computer and Systems Sciences.

Tan, T., Li, Q., Boehm, B., Yang, Y., He, M., Moazeni, R., (2009). *Productivity trends in incremental and iterative software development*. Proceedings of 3rd International Symposium on Empirical Software Engineering and Measurement. (ESEM '09), IEEE Computer Society, Washington, DC, USA.

Thomas, M., (2001). *It projects sink or swim.* British computer society review 2001.

Thong, J.y.L., Hong, S.J.,Tam, K.y., (2006*) The effects of post-adoption beliefs on the expectation-confirmation model for information technology continuance*. International Journal of Human–Computer Studies, 64, 9, 799–810.

Torrecilla-Salinas, C.J., Sedeño, J., Escalona, M.J., Mejías, M., (2015). *Estimating, planning and managing Agile Web development projects under a value-based perspective.* Department of Computer Languages and Systems, University of Seville, Spain.

Trendowicz, A., Münch, J., (2009). *Factors influencing software development productivity – state-of-the-art and industrial experiences*. Advances in Computers 77.

Vlietland, J., Vliet, H. (2014). *Towards a governance framework for chains of Scrum teams.* Department of Computer Science, VU University Amsterdam, Amsterdam, The Netherlands.

Vähäniitty, J., (2012*). Towards agile product and portfolio management*.

Williams, L., (2007). *A Survey of Agile Development Methodologies*.

Electronic references

Telia (2015). [Online] Available from: https://www.telia.se/privat/omteliasonera
[Accessed: 10 April 2015]

NE (2015). [Online] Avaliable from:
http://www.ne.se/uppslagsverk/encyklopedi/lång/effektivitet [Acessed: 4 May 2015]

Appendix

Appendix 1: Interview guide

Här nedan följer vår intervjuguide som skrevs inför de intervjuer som genomfördes på telia.se. Då semi-strukturella intervjuer genomfördes så användes endast dessa frågor som en mall och diskussionen försöktes hållas öppen kring huvudrubriken för varje kriterie. Dock togs stöd från de direkta frågorna vid behov.

Allmänna frågor:

- Vad är din position? Vad har du för befattning?
- Vilken avdelning jobbar du på?
- Hur länge har du arbetat här?
- Hur lång erfarenhet har du av att jobba agile?
- Har du erfarenhet av Telias tidigare plandrivna arbetssätt vattenfallsmetoden?
- Vad såg du för fördelar och nackdelar med att arbeta i vattenfall?
- Vad ser du för fördelar och nackdelar med att arbeta i agilt?

Personaltillfredsställelse

- Hur är din generella tanke till att arbeta agilt?
- Hur anser du att det fungerar att arbeta i agila "team"?
- Hur fungerar samarbete vid ett agilt arbete?
- Hur upplever du dina kollegors/personalens inställning till att arbeta agilt?
    - Hur är din inställning?
    - Har denna förändrats sedan telia.se implementerade agile?
- Vid ett agilt arbetssätt har team-medlemmarna fått ett ökat ansvarstagande, vad tycker du att detta har lett till?
    - Ökad prestation?
    - Ger det jobbet ökad mening?
- Har ett agilt arbetssätt lett till några svårigheter/nackdelar?

- Stress? Hektiskt? Rörigt?

- Ser du några fler fördelar att jobba agilt?

- Många argumenterar för att det är svårt att implementera agile. Hur upplevde du att denna process var?

Lärande processen

- Har teamens produktivitet förändrats nu sedan den agile implementationen?
- Hur upplever du att kunskapandet har förändrats inom organisationen sedan agile implementerades?
- Har kreativiteten förändrats sedan implementering av agile?
- Hur upplever du organisationens disciplin?
- Hur upplever du team medlemmarnas disciplin?
- Det argumenteras för att det finns en bristfällig dokumentation inom det agila arbetssättet och att detta kan leda till en mindre organisationshistoria, vilket kan leda till mindre förståelse i det långa perspektivet Vad är din synpunkt på det hela?

Team produktivitet

- Upplever du att teamens produktivitet har förändrats?
- I så fall på vilket sätt?
- Finns det förbättringsmöjligheter?

Kvalité på dagens lösningar

- Hur upplever du kvalitén på dagens lösningar?
- Hur upplever du kundtillfredsställelsen med dagens lösningar/funktionaliteter?
    - Är det bättre nu än tidigare?
- Hur många utav era lösningar går ut i release/omarbetas?
- Många argumentera för att man upptäcker buggar/fel i era lösningar tidigare nu, vad är din synpunkt på detta?
- Några argumenterar för att lösningarna i agile kan bli fört kortsiktigt inriktade, vad är din synpunkt på detta?

Kundnöjdhet

- Hur nöjda är kunderna generellt med dagens lösningar?
- Hur fungerar dagens kundrelationer?
- Hur mäter ni customer satisfaction? Har denna förbättras?

Time-to-market

- Får ni ut funktionalitet snabbare på marknaden nu?
    - Vad tror du detta beror på?
- Vad är din åsikt angående att agile uppmuntrar ett bättre proaktivt beteende i en förändrande värld?

Kommunikation

- Hur tycker du att kommunikationen mellan avdelningarna fungerar?
    - Är den bättre eller sämre nu?
- Har IT-avdelningen och Business avdelningen samma prioriteringar när det gäller era lösningar?
    - Söker avdelningarna efter samma krav på lösningar?
    - Snabba lösningar vs kvalite?
- Hur anser du att kommunikationen mellan team medlemmarna fungerar?
- Hur anser du att kommunikationen/koordineringen mellan team fungerar?

Om du skulle vilja ge ett råd till telia.se angående den fortsatta implementeringen av agile software developement, vilken förändring eller förbättring hade du velat genomföra?

Appendix 2: Respondents

Maria Lund

Lund works as a solution expert and business analyst. She is a support to the product owner with expertise and analysis in the agile working method. She has been working at TeliaSonera since 1985.

Thomas Alm

Alm is responsible for the entire department, it involves personnel, budgetary responsibility, and also that they deliver and operate the projects and missions that they are involved in at Telia.se. He are also responsible for several other portals within Telia such halebop.se and Skanova.se. Alm has worked at TeliaSonera for thirteen years at various occasions.

Beatrice Düring

Düring works as an agile coach at telia.se, which means that she helps telia.se with the agile deployment process. She describes her work as a "transformation coach", as a major organizational cultural change must take place. She is hired as consultant and has worked at TeliaSonera since 2009.

Thomas Trolltoft

Trolltoft works in the development lead, which means that he is responsible for a development team with system developers and testers. He delivers customer orders and then plans them into the releases and making sure it gets done. Trolltoft is also involved in developing technical solutions and he has worked at TeliaSonera for five years.

Quintus Lindblom

Lindblom works as a product owner at telia.se. His responsibility is support and customer service. His goal is to create a good online experience for Telias customers. He works at the online department and has worked at TeliaSonera since 2007.

Mikael Eriksson

Eriksson's official title is IT-architect, but he is responsible for resource allocation. Furthermore, he is responsible for operating efficiency at telia.se. He works at the IT department and has worked at TeliaSonera since 1996.

Martin Hedbäck

Hedbäck works in the development lead, which means that he is responsible for a development team that program functionalities. He works at the IT department and has worked at TeliaSonera for 6 years.

Anders Meyer

Meyer is head of Online Sweden. He is managing all Online channels within Telia Sweden. He is responsible for the product owners as he works as chief product owner. He works at the digital channels department and he has worked at TeliaSonera for seventeen years.