



LUNDS UNIVERSITET

Ekonomihögskolan

Institutionen för informatik

Risk Management in Agile Projects

Integrating aspects of risk management with agile methods

Kandidatuppsats 15 hp, kurs SYSK02 i informationssystem

Författare: Douglas Erol Nilsson

Handledare: Odd Steen

Examinatorer: Markus Lahtinen
Benjamin Weaver

Risk Management in Agile Projects: Integrating aspects of risk management with agile methods

Författare: Douglas Erol Nilsson

Utgivare: Inst. för informatik, Ekonomihögskolan, Lund universitet

Dokumenttyp: Kandidatuppsats

Antal sidor: 102

Nyckelord: Software Development Methodologies, Agile methods, Risk mitigation, Agile risk management, Factors and benefits of agile risk management

Sammanfattning (Max. 200 ord):

The necessity of agile risk management has been a debated topic. While some claim that agile methods mitigate risks with their agile techniques and principles, others claim that it is not enough and an explicit risk management process is necessary. The findings of this study show that agile methods do mitigate risks extensively. By working closely with customers, following an iterative and incremental development process and by utilizing certain agile techniques, agile methods are able to fulfil similar purposes that risk management phases do. This study also implies that while agile methods effectively mitigate risks, there are certain benefits with integrating aspects of risk management such as avoiding pitfalls, better understanding requirements and planning the project. However, these benefits must be compared with the value generated by other tasks in the backlog. Factors such as the project size, complexity and the customer are found to be the main factors affecting the necessity of agile risk management. Despite the lack of an explicit risk management process in agile methods, the study found that the investigated companies carry out certain aspects of agile risk management through pre-project planning and treating risks like regular backlog items.

Table of Contents

1	Introduction.....	1
1.1	Background.....	1
1.2	Research Problem.....	2
1.3	Research Question.....	2
1.4	Purpose.....	2
1.5	Scope.....	3
2	Literature Review.....	4
2.1	Traditional Methods.....	4
2.1.1	Commonalities of Traditional Methods.....	4
2.1.2	Shortcomings of Traditional Methods.....	4
2.2	Agile Methods.....	5
2.2.1	Iterative and Incremental Development.....	5
2.2.2	The Main Principles of Agile Methods.....	6
2.2.3	Risk Mitigation with Agile Methods.....	7
2.3	Risk Management.....	9
2.3.1	An Attempt to Increase Project Success.....	9
2.3.2	Traditional Risk Management Phases.....	9
2.4	Agile Risk Management.....	11
2.4.1	The Need for Risk Management in Agile Methods.....	11
2.4.2	Integrating Risk Management with Agile Methods.....	13
2.4.3	Artefacts and Tools.....	15
2.5	Research Model.....	16
2.5.1	The Agile Method.....	17
2.5.2	ARM.....	18
2.5.3	The Necessity of ARM.....	18
3	Research Methodology.....	20
3.1	Choice of Method.....	20
3.2	Selection of Participants.....	20
3.3	Interview Guide.....	21
3.4	Data collection and analysis.....	22
3.5	Reliability and Validity.....	22
3.6	Ethical Aspects.....	23

4	Empirical Findings.....	25
4.1	The Agile Method.....	25
4.1.1	Risk Mitigation with Agile Methods.....	25
4.2	Agile Risk Management	28
4.2.1	Integrating Risk Management with Agile Methods	28
4.2.2	Use of Artefacts.....	30
4.3	The Necessity of ARM	31
4.3.1	Effects of ARM	31
4.3.2	Factors Affecting the Necessity of ARM.....	32
5	Discussion	35
5.1	The Agile Method.....	35
5.1.1	Risk Mitigation with Agile Methods.....	35
5.2	Agile Risk Management	36
5.2.1	Integrating Risk Management with Agile Methods	36
5.2.2	Use of Artefacts.....	38
5.3	The Necessity of ARM	38
5.3.1	Effects of Risk Management	38
5.3.2	Factors Affecting the Necessity of ARM.....	39
6	Conclusion	42
7	Appendices.....	43
7.1	Interview Request Template	43
7.2	Interview Questions	44
7.3	Interviews	46
8	References.....	93

Figures

Figure 1: Agile risk management process (Moran, 2014, p. 38)	13
Figure 2: Overview of ARM practices and artefacts (Moran, 2016)	166

Tables

Table 1: Impact of agile methods on software development risks (McConnell, 1996; Albardaneh et al., 2015)	8
Table 2: Themes of the research model	17
Table 3: Selected participants and their experience	21
Table 4: Summary of project details	32

1 Introduction

1.1 Background

Until the 1960s, software development followed no framework or methodology. Software projects often depended solely on the competency and skills of the programmers. They were often following the rule of the thumb. The information systems they developed were not always meeting the customer requirements. They were often behind schedule and over budget. If customers desired to make changes to the information system once the development process had begun, complications were unavoidable. These factors resulted in customers being displeased by what they received in return for their investment in IT. Project failure was a most likely event (Avison & Fitzgerald, 2006).

In the light of such factors and the growing need for complex information systems, customers started to value analysis, design and methodologies for software development. They were willing to pay more, and following these values increased the likelihood of delivering a desirable product (Avison & Fitzgerald, 2006).

Organizations working with software development heeded what the customers valued and started working with Software Development Methodologies (SDM). Amongst the first methodologies to be implemented came traditional, plan-based methodologies. Traditional risk management phases have been used in these SDM (Sommerville, 2016). Since the adoption of the first SDM, there has been new ones being implemented and the existing ones have been going through innovation (Avison & Fitzgerald, 2006).

While there are many reasons for the change and development of SDM, risks have always played a central role in it. In an ever-changing world, the demands on a project are inevitably subject to change, which poses risks. The traditional methods were lacking when it came to meet the changing demands and requirements, resulting in less successful IT projects (Sommerville, 2016). In order to adjust to the new circumstances, iterative and agile methods were introduced as an alternative. Agile SDM were able to manage the risks that have been arising in traditional methods (Coram & Bohner, 2005; Sommerville, 2016).

However, agile methods are not silver bullets which solve all the problems with software development and projects (Fowler & Highsmith, 2001). Like all other project management and SDM, there are risks associated with, or which arise in agile methods as well. One way to mitigate these risks and to increase the chances of project success, is to integrate risk management processes with the agile methods (Coram & Bohner, 2005; Gold & Vassell, 2015).

1.2 Research Problem

Agile methods started to be adopted in a wider scale after the agile manifesto was issued. Agile methods were designed to meet the changing demands of customers through iterative deliveries where customers would be given the chance to observe and be a part of the development of their desired software. Thereby, the customers could give feedback and change their requirements. With agile methods, more focus is laid on the customer and the development of the product rather than strictly following processes (Cohen et al., 2004).

Due to their ability to adapt to changing requirements and circumstances, agile methods have been known to mitigate risks such as meeting the customer demands, staying in budget and time, which are common threats to traditional methods. Given its adaptive nature, it can be interpreted that agile methods do not require any risk management whatsoever. It is often presumed that agile methods have risk mitigation processes within the method. Thus, there is a belief that agile methods manage risks implicitly. Whether additional risk management is necessary in agile methods is an ongoing discussion (Walczak & Kuchta, 2013; Gold & Vassell, 2015).

Gold and Vassell (2015) claim that integrating agile methods and risk management processes would balance the methods and increase the odds of project success. Khatri et al. (2014) claim that every SDM is prone to risks, and there should be risk management involved in any SDM to identify and assess them as well as plan a counter-measure. According to Nyfjord and Kajko-Mattsson (2007), agile methods promise to be risk driven yet they do not provide any suggestions as to how risks should be managed.

There appears to be two contradicting perspectives on risk management in agile projects. The first one states that agile methods do not require risk management, since the method was made to mitigate risks and it can adapt to changing circumstances. The other implies that agile methods alone are not able to manage risks entirely, therefore integrating an explicit risk management process would be beneficial for agile methods.

Moreover, to what extent risk management is integrated in agile methods and how risk management ought to be carried out in agile methods appears to be an obscure field which will be further investigated in this study.

1.3 Research Question

Which aspects of risk management are implemented in agile projects?

1.4 Purpose

This study aims to investigate how risks are managed in agile software development projects, and what aspects of risk management are used to achieve this. To understand how risk

management has an impact on the project. To assess whether and to what extent agile methods implicitly mitigate risks enough to eliminate the need for explicit risk management.

1.5 Scope

For the purposes of this study, risk carries the following meaning: “*an uncertain event or condition that, if it occurs, has a positive or negative effect on a project’s objectives*” (PMI, 2009, p. 4). Given this definition, this study focuses on risks which have a negative effect since risk management is more commonly used to handle such risks. This study aims to investigate how risks are managed in agile projects, and to what extent agile risk management is integrated with this process. To do so, this study focuses on individuals with experience and knowledge within agile methods and risk management. Therefore, the chosen interviewees are or have been project leaders in agile projects, and were found from IT consultancy companies.

2 Literature Review

2.1 Traditional Methods

To understand the risk mitigating nature of, and the wide scale transition to agile methods, it is necessary to have some prior knowledge on the earlier methods and their flaws.

2.1.1 Commonalities of Traditional Methods

Amongst the first SDM to be adopted came traditional, plan-based methods (Avison & Fitzgerald, 2006). They rely on thoroughly planning the phases of the project at an early stage and rigorously controlling the software development process (Sommerville, 2016). The most common traditional method is the waterfall model. Therefore, the example below is mainly based on the waterfall model, though similar phases are applied to other traditional methods.

In the waterfall model, software development is carried out in a series of sequential phases. One phase must be completed before the next one could be fully in progress. The *requirement analysis* phase determines the user stories, tasks and how the other phases will be executed. Once the initial phase is complete and a plan is established, the *design* phase defines which components will be developed and the architecture behind the end product. Development starts in the *implementation* phase, where testing is also carried out. Once developed, the product is integrated into the customer's system and tested during this *verification* phase to ensure that there are no flaws or irregularities. After the product is delivered, the *maintenance* phase commences, where the product is kept under surveillance and assistance is provided to the customer in form of updates or fixing errors (Sommerville, 2016). A *documentation* is created at the end of each phase describing the requirements and objectives of the respective phase (Munassar & Govardhan, 2010). If changing the plan is necessary, the stakeholders must be informed and the new plan must be confirmed (PMBOK, 2016).

Traditional methods are often preferred by large organizations which work with teams distributed across geographical locations. Traditional methods have been and continue to be useful when the plan established at the beginning of the project does not change (Sommerville, 2016). When the requirement is to develop large information systems, systems which require a high level of security or involve embedded systems (Sommerville, 2016). Traditional methods are useful when the project team members lack expertise and instead can follow and rely on a plan. Traditional methods are also relatively easy to implement and follow (Munassar & Govardhan, 2010).

2.1.2 Shortcomings of Traditional Methods

Traditional methods often do not fit well where the circumstances of software development changes. In the 1980's, it was believed that rigorously planning, analysing and designing for a software project would lead to success (Sommerville, 2016). Yet times have changed, and

customers rarely are certain of what they desire or require from the beginning of the project. If they do, these requirements are most likely to change (Sommerville, 2016).

Following a sequential order does not give room for change (Sommerville, 2016). Once the project has commenced, returning to the previous phases could lead to the repetition of some phases, increasing the costs and delaying the delivery of the project. For example, the implementation phase starts after a comprehensive amount of planning and designing. When errors or limitations are discovered in the implementation phase, the requirements and design may have to be changed and repeated (Munassar & Govardhan, 2010).

In addition, traditional methods put excessive time into documentation. The administrative costs for traditional methods make them inappropriate for small projects (Munassar & Govardhan, 2010).

While this way of working with software projects was functioning well with most projects in the 20th century, dissatisfaction with traditional methods increased in the 1990's (Sommerville, 2016). The main reason for this is noted to be the advancements in technology, and the rapidly changing circumstances and requirements (Whitney & Daniels, 2013). Traditional methods started to fail in meeting their requirements. Projects were either exceeding their budget, falling behind schedule or delivering a product out of scope (Patton, 2002). "Dissatisfaction with these heavyweight approaches to software engineering led to the development of agile methods in the late 1990's." (Sommerville, 2016, p. 75).

2.2 Agile Methods

Agile methods were adopted as a solution to the major problems faced in traditional methods. In this section, the essentials of agile methods are explained while also focusing on how they mitigate risks.

2.2.1 Iterative and Incremental Development

Agile methods are based on iterative and incremental methods. Such methods can be dated back to Lean development in the 1990's, Rapid Application Development in the 1980's, Joint Application Development in the 1970's or even further back in time (Jiang & Eberlein, 2009). The values and principles of agile software development has existed for decades.

However, in 2001, the Agile Manifesto was brought forth by experienced developers whom advocated a different way of how software should be developed. Frustrated by the solid struck processes of traditional methods, they proposed a change to the priorities of software engineering by writing a manifesto (Chow & Cao, 2008).

The agile manifesto prioritizes four main values:

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

“While there is value in the items on right, we value the items on the left more.” (Chow & Cao, 2008, p. 962). Although they may differ slightly in practice, agile methods share the same ideologies and strive to follow the values stated in the agile manifesto (Lindvall et al., 2002).

The values of the agile manifesto aim to optimize the software development processes and the structure of development teams (Elbanna, 2016). The agile values suggest that rather than prioritizing processes which may lead to undesirable results, the individuals involved in the project should have a more central role (Cohen et al., 2004).

Unnecessary work and processes should be minimized, such as creating certain documentations which would never be used (Dingsøyr et al., 2012). Instead, focus ought to be laid on developing the software which the customer needs. The skills of the developers and timely feedback from the customer is more likely to lead to project success in comparison to blindly following a project lifecycle (Avison & Fitzgerald, 2006). This requires customer collaboration and adapting to changing requirements (Cervone, 2011).

These values are mainly followed by basing the software development method on iterative and incremental cycles. In these cycles, backlog items are completed and tested before integrating them. The product is demonstrated to the customer, and feedback is received to make necessary changes in the next iteration or increment (Lindvall et al., 2004).

The customers are frequently convened, enabling the development of the end product with respect to the customer’s needs. The needs of the customer and the developers are bound to change, yet through constant feedback from both sides, changes can be accommodated to the end product (Dingsøyr et al., 2012; Elbanna, 2016). Iterative development improves team collaboration through facilitating communication and preventing disruptions in the project. This leads to the delivery of the needed product in a rapid manner (Cervone, 2011).

2.2.2 The Main Principles of Agile Methods

Customer involvement: The central idea behind agile methods is delivering customers what they want and need. This requires a close contact with the customer. There must be a constant communication in progress between the customer and the project team. Thus, the customer can request changes to the software, the customer can be consulted about what to prioritize and both parts keep each other updated. The lack of customer involvement tends to result in the development of products which are not what the customer requested or needs by the time of delivery (Lindvall et al., 2002; Dingsøyr et al., 2012; Sommerville, 2016).

Embrace change: Circumstances will change, and so will the customer's requirements. There may be new functionality requests, the priorities may change or there may be defects with the system. In any case, agile methods should embrace change and manage to adapt to the changes (Henninger et al., 2002; Sommerville, 2016).

Incremental delivery: Agile methods develop software in an iterative and incremental manner. At each increment a new set of requirements are chosen to add to the software, further improving the end product. Often the functionality and time that will be put into the increment is not changed while it is in progress. The customers are involved to get rapid feedback during the development process. New requests or changes are taken into consideration for the next increment. By the end of each increment, a functioning software is attained (Alleman, 2002; Sommerville, 2016).

Maintain simplicity: Agile methods strive to escape from the complexities of traditional methods. Both in terms of the software being developed, and the process of developing it. Time demanding yet vital processes such as testing, integration or configuration are eased with the use of tools and automation (Alleman, 2002; Sommerville, 2016).

People, not processes: Agile methods rely on the skills of the development team. There is a significant level of trust and responsibility laid upon them. In agile methods, they are more independent and they do not follow a determined process. The team is able to organize itself to complete objectives (Sommerville, 2016).

Communication and documentation: Informal communication mediums are preferred over detailed and cumbersome documentations. Documentation is minimized and mostly consists of the information about the necessary characteristics of the product and the user requirements in the form of an outline. Despite being a poor choice of communication, documentation is necessary to keep and pass on vital information (Lindvall et al., 2002).

2.2.3 Risk Mitigation with Agile Methods

Traditional methods often faced risks such as customer dissatisfaction, feature creep or unrealistic scheduling and budgeting (McConnell, 1996). Agile methods were primarily adopted because they were known to mitigate risks which arose in traditional methods, as shown in Table 1.

<i>Common Risks in Software Development</i>	<i>Impact of Agile on the Risks</i>
<i>Scope creep (undesirable growth of the project scope)</i>	Reduce
<i>Overly optimistic schedules</i>	Reduce
<i>Requirements or developer gold-plating (adding unnecessary functionality)</i>	Reduce
<i>Shortchanged quality</i>	Reduce
<i>Inadequate design</i>	Possibly increase
<i>Silver bullet syndrome</i>	Increase
<i>Friction between developers and customers</i>	Reduce
<i>Weak personnel</i>	--
<i>Contractor failure</i>	--

Table 1: Impact of agile methods on software development risks (McConnell, 1996; Albardaneh et al., 2015)

Agile methods comprise several characteristics which are believed to mitigate project risks. Given its incremental and iterative nature, agile methods have an evolving list of requirements, mainly through having a close contact with the customer (Cervone, 2011). This allows better handling of shifting requirements (Lindvall et al., 2004) which is where risks often arise. As Coram and Bohner (2005) says, changes made to a project towards the end will be more costly. Therefore, being able to respond quickly to changes will reduce the costs and risks of the project.

Another phase often practiced in agile methods is daily standups or meetings, where the project team gathers to share their progress and problems. This is a forum where project risks may be brought up. Being aware of potential problems and risks early on allows the team to take corrective action (Lindvall et al, 2014).

Through these attributes of agile methods, common risks with software projects which often were a threat to traditional methods are mitigated (Walczak & Kuchta, 2013). The implicit way of handling risks has led to a belief that agile methods mitigate risks throughout the project, and therefore, do not require an additional risk management process (Gold & Vassell, 2015). This has given rise to a notion that agile methods identify, tackle and restrain risks implicitly (Moran, 2014). Although agile methods are claimed to be risk-driven, there are no suggestions as to how risks ought to be managed (Nyfjord & Kajko-Mattsson, 2007). Risk management is not mentioned in the agile manifesto.

There are some who claim that agile methods do not manage risks completely on their own, therefore integrating risk management processes with the agile methods may prove to be beneficial. More of this in chapter 2.4.

2.3 Risk Management

Before going into integrating risk management with agile methods, traditional risk management and its phases are introduced.

2.3.1 An Attempt to Increase Project Success

Risk management is commonly defined as “*a set of principles and practices aimed at identifying, analysing and handling risk factors to improve the chances of achieving a successful project outcome and/or avoid project failure*” (Bannerman, 2008, p. 2120). Meanwhile, as mentioned before, the Project Management Institute defines **project risk** as “*an uncertain event or condition that, if it occurs, has a positive or negative effect on a project’s objectives*” (PMI, 2009, p. 4).

Given these definitions, every IT project has some level of uncertainty and risks. Uncertainty caused by various understandings of the requirements by the stakeholders, assumptions, constraints etc. (PMI, 2009; Sommerville, 2016). SDM and project management has been one approach to control the uncertainties associated with the projects by following a certain structured process throughout the project (PMI, 2009).

Risk management is a part of the structured project management process, which attempts to control project uncertainties through understanding, assessing and managing them. Thereby, risk management is regarded as a critical success factor (PMI, 2009). By identifying uncertainties and potential risks associated with a project, risk management contributes to the estimating, planning and designing phases of the project (Chin, 2004 & PMI, 2009). In this way, risk management contributes to project management. Therefore, it is found that it also contributes to project success (Bakker et al., 2009). Risk management emphasizes the risks which affect the critical success factors of the project. Thus, risks can be prioritized and the project can be more effectively managed by taking steps to handle the risks (Boehm, 1991; Sommerville, 2016).

2.3.2 Traditional Risk Management Phases

Although there are several risk management processes such as AS/NZS 4360, IEEE1540 or PMBoK, the phases listed below are the most renowned phases and the common denominator (Nyfjord, 2008).

Identification:

Project risks can be categorized as identified and unidentified risks. Unidentified risks are difficult to prepare for. The main approach to unidentified risks is to treat them in a reactive manner, taking measures once the threat of the risk has affected the project (Bannerman, 2008). With identified risks however, there is an opportunity to take a proactive approach.

Risks are identified in the identification phase, where a risk register is created and maintained. The identified risks are documented into the risk register together with the consequences and source of the risks (Moran, 2014; PMBoK, 2016).

Risks are identified by the stakeholders of the project, such as project leaders, project members, customers, end users, specialists etc. By brainstorming, investigating earlier projects and their risks (lessons learned), conducting interviews, carrying out root cause analysis or with the use of other techniques, the stakeholders gather information on project risks or uncertainties (Nyfjord & Kajko-Mattsson, 2007; PMBoK, 2016).

A risk register is never completed. Although the identification phase traditionally takes place when the project is initiated, the risk register should be updated throughout the project. If new risks are identified during the project, if a risk escalates or becomes insignificant, it should be documented in the risk register (Nyfjord & Kajko-Mattsson, 2007; PMBoK, 2016).

Assessment:

After the project risks are identified, they are analyzed and prioritized. A list with the prioritized risks is created, and it is validated with the stakeholders of the project. While a preliminary plan is made for the high priority risks, lower priority risks are put on a watch list. If the risks on the watch list become more severe, preliminary plans are made for these risks as well (Nyfjord & Kajko-Mattsson, 2007).

The risks can be analyzed qualitatively or quantitatively. Qualitative risk assessment is the conventional risk analysis method. The probability (likelihood) and impact of the risks are analyzed to determine an exposure rating. A prioritized risk list is created with respect to the exposure ratings. The most prioritized risks may thereafter go through quantitative assessment to determine how the risk will affect the project goals (PMBoK, 2016). Risk management should not cost more than the value it may bring to the project (PMI, 2009; Moran, 2014). By assessing risks qualitatively or quantitatively, risks which will be handled can be chosen strategically. The priorities of risks may also depend on whether the measures for them are of a proactive or reactive nature (Moran, 2014).

Planning:

Once the risks have been identified and assessed, plans and strategies can be made against the risks or a group of similar risks. The plans ultimately aim to minimize the likelihood or impact of threats posed by the project risks (PMBoK, 2016).

Plans are made for the assessed and prioritized risks. In these plans, a risk owner is assigned to the risk. A way of **controlling and monitoring** the risk with key performance indexes is determined. **Action plans** are determined to handle the risks, together with a **fallback plan** in case the action plan fails. **Contingency plans** may also be created to be prepared for risks which potentially may severely impact the project (Nyfjord & Kajko-Mattsson, 2007).

There are four main strategies for these plans. **Avoid** the threats posed by the risks by altering the project plan or project goals. **Transfer** the responsibility for the risk to a third party which is more qualified for handling or taking ownership of the risk. **Reduce** the impact which the

risk may have on the project to make it tolerable by proactively taking measures such as testing, prototyping or minimizing complex processes. **Accept** the risk if it is not feasible to act against it, but create a documentation about the risk and a reactive plan for unpredicted consequences (PMBOK, 2016).

Monitoring:

The plans that are carried out changes the status of the risk. It needs to be monitored to determine to what extent the plans have been successful, and whether further measures need to be taken. The findings of the monitoring are noted and the risk register is updated (Nyfjord & Kajko-Mattsson, 2007; PMBoK, 2016). The monitoring phase assures that an adequate amount of documentation and communication is done as well as adding accountability to the risk management process (Moran, 2014).

Risk management review:

Nyfjord (2008) emphasizes the final phase which is occasionally omitted from the risk management process. In this phase, how the risks were managed is reviewed in this retrospective phase. Evaluations are made regarding which risks were identified, what were their causes, how were they treated, what were the results. In this way, there can be lessons learned for future projects. The risk register is updated with the latest status of the risks and someone signs off the risks, thus taking responsibility for the results.

2.4 Agile Risk Management

As defined earlier, traditional risk management is mainly based on identifying, assessing, planning, handling and monitoring risks which may have a positive or negative impact on the project or organization (PMBOK, 2016). Agile risk management (ARM) involves incorporating aspects of risk management into agile methods in a way which is consistent with the agile principles (Moran, 2014).

2.4.1 The Need for Risk Management in Agile Methods

Whether agile methods need an explicit risk management process is debatable. While some claim that agile methods per se mitigate and handle risks as explained in chapter 2.2.3, others suggest that they are still subject to risks and would benefit from integrating with risk management processes (Walczak & Kuchta, 2013).

Moran (2014) as well as Khatri et al. (2014) states that every software project is prone to risks, therefore risk management is an essential part of software development methods. Although agile methods may mitigate certain risks which traditional methods were prone to, agile methods still face risks and may have different success factors (Walczak & Kuchta, 2013; Elbanna, 2016).

Moran (2014) believes that risks should be managed actively and not be left to the chance of mitigating them implicitly through the incremental and iterative agile processes. As Gilb (1988) mentioned, if risks aren't actively attacked by you, the risks will actively attack you.

Andrat and Jaswal (2015) states that certain project risks are overlooked in agile methods, therefore they remain unknown. This may be a result of having iterative and overlapping development phases rather than following a sequential order where risks are identified with one phase before commencing with the next one. Therefore, Andrat and Jaswal suggests that risk management in agile methods must be improved.

On the other hand, Nyfjord & Kajko-Mattsson (2007) justifies the absence of risk management processes in agile methods by suggesting that integrating them may be against the agile values and principles. The main criticism to their ARM model were related to their adaptability to small projects as well as discouraging collaboration and customer involvement. If risk management were to disrupt the agile principles, it may be more inconvenient than helpful to integrate them.

Nyfjord & Kajko-Mattsson (2007) points out that the need for risk management in agile methods may vary with respect to a series of factors. Hence arises the following question, "at what point do the inherent practices of agility fail to provide sufficient safeguard for projects" (Moran, 2014, p. 60). The size and scope of the project, the complexity of the product being developed, having a geographically distributed team and the attitude of the project team towards risk are pointed out to be the most important factors regarding the need for risk management (Nyfjord, 2008; Moran, 2014). For example, a simple and small-scaled project being developed by a small local team may not face significant risks which could pose a threat to the project or the organization. Therefore, some would argue that risk management would not be necessary in such a project.

According to a study done by the Project Management Institute (PMI), effective risk management increases the stakeholder confidence, the ability to respond to risks while it decreases the likelihood of taking inefficient risk measures. It also allows taking proactive measures, thus possibly reducing development costs (Nyfjord & Kajko-Mattsson, 2007). Organizational agility is enhanced by maintaining a robust change management and risk management (PMI, 2015). Moran (2014) claims that ARM helps with further handling project uncertainties by being aware of, understanding and communicating risks and the source of them.

Trusting agile methods completely and turning a blind eye to risk management can have consequences on the project such as not being able to handle risks efficiently or not identifying risks in time to treat them. Ultimately it could be one of the causes of project failure. It is believed that integrating agile methods and risk management processes would balance the methods and increase the odds of project success (Moran, 2014; Gold & Vassell, 2015).

Jiang and Eberlein (2009) claim that every project is unique, that the techniques and development processes change even when working with the same company. Since there are

no identical projects, it is unlikely to find a SDM which suits every project (Glass, 2004). Instead, the SDM need to be customized to suit the projects at hand. The same applies to ARM as Moran (2014) indicates; the situation of a project dictates how it will be conducted. Nonetheless, certain guidelines can be created on how ARM should be conducted, as described in the next chapter.

2.4.2 Integrating Risk Management with Agile Methods

When integrating risk management approaches to agile methods, emphasis must be laid on retaining and being consistent with the qualities of agile methods (Ylimannela, 2011; Moran, 2014; Andrat & Jaswal, 2015). Staying true to the agile principles would retain transparency and effective communication as well as hasty decision making which are essential for agile methods (Moran, 2014).

While there are multiple studies on the need for integrating risk management with agile methods, not many studies have been found on *how* it should be done. There are several suggestions and case studies focused on specific companies, but in this theoretical study, none except Moran's work was found to be comprehensive and apply on a general scale. The phases below are introduced by Moran (2014) and simplified in Figure 1, but some can also be observed in the case studies of other authors.

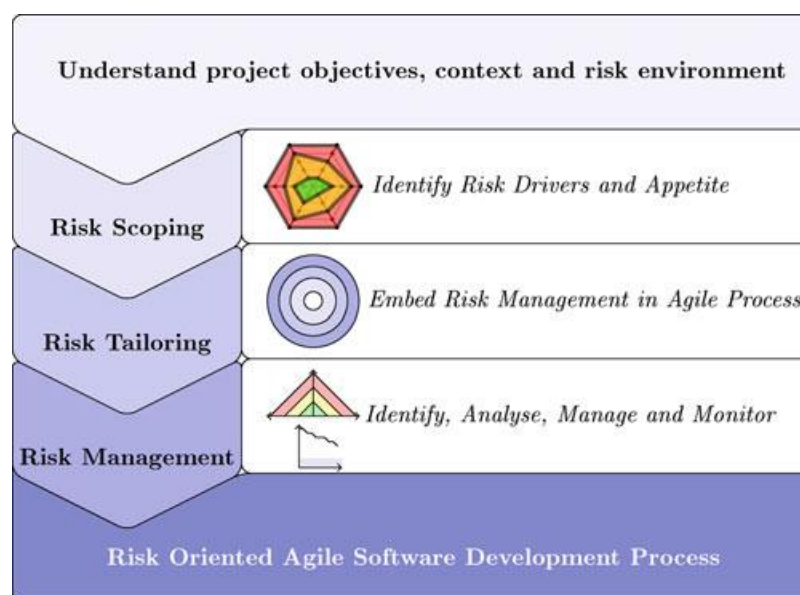


Figure 1: Agile risk management process (Moran, 2014, p. 38)

Project Context and Risk Environment:

A phase which occurs initially to put every team member on the same page. The project and the requirements are analysed to determine a common attitude towards risks and to assess which level of risks the project could manage. This requires that the project goals are known.

Being aware of the basic project risks, such as risks regarding the requirements, technicalities, suppliers, schedule etc. will also provide a basis on which the proposed project can be assessed. The assessed basic project risks are charted and used to judge whether or not the project is feasible to carry on with (Moran, 2014). Nyfjord (2008) describes a similar process, *Product Vision Planning*, to identify business related risks which may concern the organization. This is done by assessing the vision, goals and return of investment of the project.

Risk Scoping:

In this phase, how the basic project risks will affect the project specifically is assessed. This should be done by the team members and primary stakeholders of the project. Through this assessment, a judgement could be made on whether there are conditions which should be met prior to commencing the project. Conditions such as allocating the necessary resources or finding potential suppliers. The project may be conducted in different ways depending on the risks it faces (Moran, 2014).

Risk Tailoring:

Once the risks have been understood in the risk scoping phase, some adjustments can be made to the agile method. Risk management activities are discussed and the most appropriate moments to carry out these activities are decided. For example, some activities such as risk reviews may have to be carried out daily on certain projects, and once per iteration in other projects. The activities may also consist of techniques used in agile methods, such as prototyping, retrospective sessions or refactoring (Moran, 2014). Nyfjord (2008) describes this process as *Product Roadmap and Release Planning*, where a plan is made for delivering the product. In this plan, risk management processes are carried out repeatedly.

Risk Management:

The phases up to this point may be considered as pre-project phases, while the risk management step takes place during the project. The aim of the risk management step is to improve the understanding of risks while also support decision making within the context of the project. The traditional risk management phases are carried out in this step.

A risk register should be kept and updated throughout the risk management process. Moran (2014) suggests that risks should be identified whenever a new iteration starts and monitored throughout the iteration. The larger and more complex the project is, the bigger the need for analysing, tracking and managing its risks.

Risks in the risk register may not have an owner. The risks should instead be treated like tasks in the backlog, where someone will eventually be assigned to it and take responsibility for it. If the risk is a larger task, multiple people can be assigned to split the responsibility and treat it.

Nyfjord (2008) on the other hand depicts various groups which ought to own and be responsible for specific types of risks even though the responsibility may shift between these

groups. The groups Nyfjord suggest takes care of risks in four levels; organizational, business, product and development level.

2.4.3 Artefacts and Tools

In order to ease the assessment and monitoring of risks, ARM tends to make use of artefacts. Artefacts are tools which visualize the circumstances and communicate the risks effectively to the project team. Artefacts also increase transparency within the project. Thus, risks can be assessed, planned for and monitored by the project team and proactive or reactive measures could be taken more wisely (Moran, 2014).

Some of the artefacts used in ARM are provided below:

Risk register: Also referred to as risk list or risk log, risks which have taken place in similar projects and risks which are identified with the current project are noted into the risk register. As the project progresses, new risks are identified by the project team and added to the risk register while the existing ones are updated. It is mainly used to keep track of identified risks and have an understanding for them (PMBOK, 2016).

Risk Burndown chart: The principles of a burndown chart from Scrum is applied to risk management. A burndown chart is used to track the progress of the project in regard to the project iterations and the completion of objectives. Similarly, a risk burndown chart assesses the presence and exposure level of risks as the project progresses (Moran, 2014).

Risk pyramid: Similar to risk matrices, a risk pyramid is used to plan which measures to take against risks. Each risk is given a value representing the likelihood of the risk, and the impact the risk would have on the project or organization. Depending on these two values, the risk is assigned an exposure value. The exposure value is used together with the risk pyramid to choose an appropriate reaction to the risk such as avoiding, transferring, reducing or accepting the risk (Moran, 2014; PMBOK, 2016).

Risk modified Kanban board: Although Kanban is a method by itself, a Kanban board can be used for risk management purposes. The Kanban board is visible to the project team and it acts as a backlog for risks, where tasks are assessed and given a risk treatment plan. This helps with the prioritization of tasks and risk planning (Moran, 2014; Lei et al., 2017). An ARM model which Ylimannela (2011) suggested was mainly adopting a risk modified Kanban board where the development team is following it throughout the project.

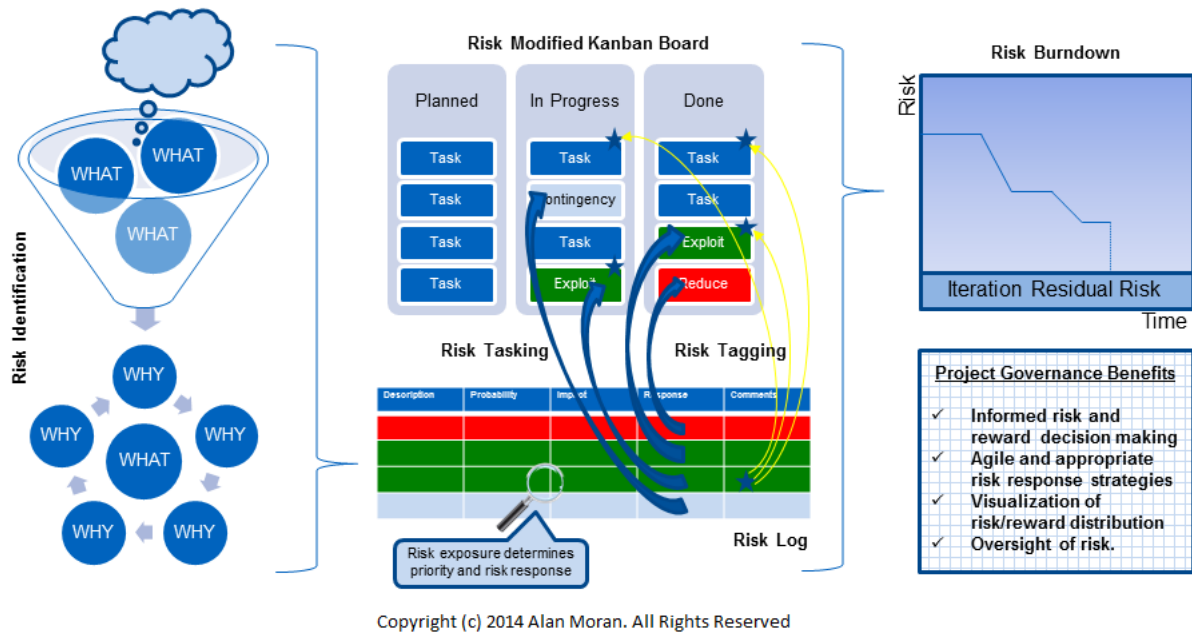


Figure 2: Overview of ARM practices and artefacts (Moran, 2016)

2.5 Research Model

With respect to the literature review, the themes and sub-themes shown in Table 2 have been chosen to attempt to answer the research question and fulfil the purpose of this study.

By looking into how the agile methods are implemented; how risks are handled with agile methods or risk management; reflecting over the necessity of ARM, this study attempts to resolve which aspects of risk management tend to be implemented in agile projects.

The themes, predominantly relevant references and the interview questions related to them are represented in the table below. Thereafter, the sub-themes which are covered in the literature review are summarized.

Theme	Sub-theme	References	Interview questions
The Agile Method			
	Risk mitigation with agile methods	McConnell, 1996 Cohen et al., 2004 Lindvall et al., 2004 Coram & Bohner, 2005 Chow & Cao, 2008 Cervone, 2011 Dingsøy et al., 2012 Walczak & Kuchta, 2013 Gold & Vassell, 2015 Sommerville, 2016	2, 3, 4, 5, 10
ARM			
	Integrating risk management with agile methods	Nyfjord, 2008 Ylimannela, 2011 Moran, 2014 Andrat & Jaswal, 2015	3, 4, 5
	Use of artefacts	Lei et al., 2007 Ylimannela, 2011 Moran, 2014 PMBok, 2016	5, 6
The Necessity of ARM			
	Effects of risk management	PMI, 2009 Moran, 2014 PMI, 2015 Gold & Vassell, 2015	7, 9
	Factors affecting the necessity of ARM	Nyfjord, 2008 Moran, 2014	2, 8

Table 2: Themes of the research model

2.5.1 The Agile Method

Risk mitigation with agile methods:

Agile methods were a form of solution to risks which regularly became a threat in traditional methods (McConnell, 1996). Certain characteristics of agile methods are perceived to be risk mitigating.

Agile methods follow the values and principles stated in the agile manifesto which dictate how a software development project should be conducted (Chow & Cao, 2008). Among other things, these principles state that individuals should be trusted and prioritized over processes. This concerns both individuals from the project team, and from the customer, which requires skilled individuals. Customer collaboration should be maintained throughout the project to pave the way for the project. Having a close contact with the customer and collaborating allows the project to better understand and respond to shifting customer needs (Cervone, 2011).

The project should embrace change, and adapt to shifting requirements. Developing in an incremental and iterative manner simplifies the prioritization of requirements which may change throughout the project (Coram & Bohner, 2005). Encouraging communication within the team in the form of standups can be used to mention project risks or problems which the project team can collaboratively manage (Lindvall et al., 2004).

In the end, prioritizing simplicity, acquiring a working software and satisfying the customer's needs are the main goals (Dingsøy et al., 2004; Lindvall et al., 2004; Sommerville, 2016).

2.5.2 ARM

Integrating risk management with agile methods:

ARM is based on integrating aspects risk with agile methods and staying consistent with the agile principles. The proposed ARM process begins with understanding basic project risks before starting development. Thereafter, the basic project risks are assessed and decisions are made on how the project should be conducted with respect to the risks. A plan is also made for how and when risk management phases should be carried out throughout the project. Once the plan is established, the project commences and risk management is carried out as planned (Nyfjord, 2008; Ylimannela, 2011; Moran, 2014).

Use of artefacts:

Artefacts are tools used to effectively visualize and communicate risks to the project team. They ease the risk management phases and they make it easier to follow an ARM process (Moran, 2014). Some examples of artefacts are: risk registers (identification), risk burndown charts (monitoring), risk pyramids or matrices (assessment) or risk modified Kanban boards (planning) (Moran, 2014; PMBoK, 2016).

2.5.3 *The Necessity of ARM*

Effects of risk management:

Risk management is noted as a critical success factor (PMI, 2009). The lack of it may prevent the project team from efficiently handling risks, which may have consequences on the project goals or ultimately may be one of the causes of project failure. On the other hand, the use of risk management increases stakeholder confidence and the ability to respond to risks proactively (Gold & Vassell, 2015; PMI, 2015). Risk management increases the awareness of risks, communicates it to the project team and increases the understanding of them and their sources. Thus, it helps with avoiding possible costs or obstructions in the project (Moran, 2014).

Factors affecting the necessity of ARM:

Since it is believed that agile methods mitigate risks implicitly, integrating explicit risk management phases may be interpreted to be unnecessary in certain circumstances. Factors affecting these circumstances are mainly noted to be the size and scope of the project, the

complexity of the product being developed and the attitude of the project team towards risks (Nyfjord, 2008; Moran, 2014).

3 Research Methodology

3.1 Choice of Method

Software engineering and SDM are fields which have heavily been researched. However, agile risk management remains an area which is not fully uncovered. There is an ongoing discussion on whether agile methods need risk management or if the implicit way of handling risks in agile methods is sufficient. This study attempts to analyse the companies' take on the matter. How they deal with risks in their agile methods, which factors impact their choice, what the impact of this choice can have on the projects and the company.

For this study, a qualitative method and approach was chosen. Since the way companies implement risk management or the reasons behind it is not entirely quantifiable or easily put in a pattern, a qualitative approach was preferred above a quantitative study. As Jacobsen (2002) points out, a qualitative study is to prefer with explorative studies which aim to analyse the topic from different point of views. The raw data which is collected from the interviewees is complex and unstructured, it demands time and analysis to make use of it (Jacobsen, 2002). Therefore, the study was conducted on a few knowledgeable individuals. How many interviews should be conducted for the study is a question with an indefinite answer. Alvehus (2013) as well as Kvale and Brinkmann (2009) states that it generally depends on the number of interviews required to reach the threshold where the same information is obtained from the interviews. Alvehus (2013) also suggests that the number of interviews may depend on the requirements from the university institution for a bachelor's thesis. Given the suggestions from the institution and the focus group of this study, five interviews were regarded to be sufficient to acquire the necessary empirical data.

3.2 Selection of Participants

In order to study which approach companies have towards agile risk management, the interviews were conducted on people from different companies. Interviewing people from the same company could have generated similar answers due to a company specific approach to risk management. To eliminate such a possible outcome, each interviewee was chosen from a different company. Thus, a heterogeneous selection of participants, in the sense of the companies they work for, was preferred. A heterogeneous approach according to Alvehus (2013) is when the selected participants are not exactly similar. This gives a wider insight as well as a richer nuance to the study.

As the literature review was nearing its completion, the search for interviewees commenced. The main approach used to find the interviewees was through face-to-face interaction. People

with a higher position in the companies, such as regional managers or leaders in IT consultancy companies were spoken with. The focus and aim of the study was explained to them, and they recommended experienced and voluntary project leaders or managers from their organization.

The chosen companies are IT consultancy companies, delivering and working on projects in an agile manner. The main criteria which was considered while searching for interviewees was their experience and knowledge regarding agile methods and risks. The experience and knowledge requirement led to a strategic selection of participants (Alvehus, 2013).

Interviewee 1 (I1)	Regional manager. Has been a developer and a project leader in agile software development projects. Experienced with the agile method.
Interviewee 2 (I2)	Digital strategist and project manager. Started working as a business developer on the customer side. Thereafter took upon the role of project manager. Has been working as a project manager for thirteen years.
Interviewee 3 (I3)	Agile project manager, occasionally a traditional project manager, sometimes a Scrum master, agile coach. Has a background in development and continues to occasionally work with development. Has been a project manager since 2010.
Interviewee 4 (I4)	Started working as a developer, has been a project manager for agile projects for three and a half years.
Interviewee 5 (I5)	Project manager and requirement analyst. Has been working as a developer and an architect. Has been working more and more with project management. Has been a project manager for almost four years.

Table 3: Selected participants and their experience

3.3 Interview Guide

Based on the literature review, ten interview questions with sub questions were listed to follow during the interviews (See 7.2 Interview Questions). According to Jacobsen (2002), the form of an interview varies between completely structured and completely open interviews. Structured interviews have all the interview questions prepared and do not go astray from these questions. While this could make it easier to compare each interview, like a survey, it may fail to extract as much valuable information as possible from the interviewees since the questions would not change depending on how the interviews progress. Open interviews on the other hand are not planned. The interviews are conducted based on what the interviewer deems fit to ask at that moment. While open interviews are flexible and adapt to each interview and interviewee, it requires that the interviewer is experienced and knowledgeable in order to extract the relevant and useful information. Jacobsen (2002) also mentions that an approach in between these two interviews can be used. Semi-structured interviews are a

mixture of both previously named interview forms. Given its possibility to adjust, semi-structured interviews were conducted for this study.

Before each interview, the interviewees were informed about the purpose and scope of the interview via mail. They were also informed that they will be anonymous in the study, and that with their consent, the interviews would be recorded. Choosing to conduct the interviews face-to-face would ease the discussion of possibly sensitive information in comparison to telephone interviews (Jacobsen, 2002). The interviews were to be conducted in the interviewees' offices, where the interviewees would be in an environment they are used to. According to Jacobsen (2002), this is an important aspect to extract information from the interviewees with the minimal effect from the environment.

Each interview started with a presentation of myself and the study once again. While there is an outline of interview questions, there is room for follow-up questions depending on the answers of the interviewees. The interview outline was not followed in the exact same order since the interviews progressed in different ways. However, each question was raised during the interviews. As Jacobsen (2002) advises, the topics are opened depending on how the interview progresses.

3.4 Data collection and analysis

The empirical data was collected by recording the interviews and transcribing them. The transcriptions follow a question and answer pattern. Each question and answer was given a respective number for easier referencing.

Jacobsen (2002) suggests that interviews should not be shorter than 30 minutes since depending on the study, not enough information may be collected from the interviews. Nor should they be longer than 90 minutes, since this may lead to information overload or a lack of focus. For this study, the length of the interviews was between 30-60 minutes. Within this timeframe, the interviewees were able to answer the questions to a satisfactory level.

Maxwell (2005) claims data analysis should be begun with right after the first interview. Therefore, after each interview, the recordings were listened to and notes were taken in order to extract key takeaways. Thus, the interviews were analysed holistically. The aim with qualitative research methods is to breakdown the data into categories or general themes for purposes of comparison and developing theoretical concepts (Maxwell, 2005). This was achieved while comparing the notes and identifying commonalities and categories to compare the interviews.

3.5 Reliability and Validity

Methods used while conducting the study and research can have an impact on the results of the study. Therefore, presenting the used methods and justifying them is integral to criticize

the results (Jacobsen, 2002). Criticism towards the quality of a study is often based on its validity and reliability.

Jacobsen (2002) divides validity into internal and external validity. Internal validity represents to what extent the study manages to fulfil its purpose, whether it has measured what it intended to measure. The informants play a central role in this. For this qualitative study, interviews were conducted with experienced and knowledgeable project leaders and managers within agile projects from different companies. As Jacobsen (2002) states, experienced informants who are working closely with the topic at hand are relatively more trustworthy. Although it cannot be guaranteed whether the interviewees are telling the truth, similar answers given by different interviewees who are not related to each other makes the results more valid (Jacobsen, 2002). Moreover, the informants accepted to be interviewed, showing their will to participate in the study.

External validity on the other hand, represents to what extent the study is generalizable. As Maxwell (2005) and Jacobsen (2002) states, the purpose of qualitative studies is often not to generalize. Instead they focus on understanding and analysing a certain topic or phenomenon (Jacobsen, 2002). The interviewees were selected to understand how they and their companies manage risks in agile projects, but this does not represent how others do it. However, it is possible to make a theoretical generalization (Jacobsen, 2002). Maxwell (2005) suggests that qualitative studies generalize based on a theory which could be applicable to a larger group. For this study, the motives behind implementing agile risk management may be generalized, but not how it is implemented.

Reliability concerns to what extent the study is repeatable and whether the same results would be obtained (Alvehus, 2013). To what extent the study was conducted in a reliable manner (Jacobsen, 2002). Both Jacobsen (2002) and Alvehus (2013) imply qualitative studies are difficult to replicate due to factors such as the time when the interviews were conducted or the relationship between the interviewer and the interviewees. However, to make the results reliable, certain steps were taken. The interviewees were informed about the context and purpose of the study, and a time and place was scheduled for the interview. Jacobsen (2002) states that the environment where the interview is conducted has an impact on the answers the interviewees give. Therefore, the interviews were conducted in their work environment where they would feel more at home. The interviews were also recorded to make the data analysis easier and present the raw data to make the results more reliable (Jacobsen, 2002).

3.6 Ethical Aspects

Conducting an ethical study may contradict with conducting a detailed study. Therefore, when studying a sensitive issue, there is a dilemma between ethics and research quality (Kvale & Brinkmann, 2009). To avoid the dilemma, a series of ethical rules are often followed. Since this study was focused on risk management, it did not involve highly sensitive information.

Nonetheless, the ethical rules were followed to respect the interviewees' privacy rights while also striving to improve the quality of the study.

The main ethical rule, informed consent, implies that the interviewees ought to be willingly participating in the study. They should be informed about the study and the purpose of the interview as well as how the collected data will be used (Jacobsen, 2002; Kvale & Brinkmann, 2009). Then, the interviewees can consider what risks or benefits may be associated with their participation and decide whether to participate or not.

During this study, all the interviewees received the same information about the purpose of the interview through mail and the same information was repeated to them before the interviews as a reminder. With their consent, the interviews were recorded and they were informed that the interviews would be transcribed and used for the study.

Another important rule is confidentiality. The privacy of the individuals ought to be respected and taken into consideration while conducting the study (Jacobsen, 2002). Since the study does not concern the interviewees' personal data, it was excluded from the transcriptions as Kvale and Brinkmann (2009) advises. For example, the interviewee, customer and company names were not included in the transcriptions to eliminate the likelihood of identifying the involved individuals. The interviewees were informed prior to the interviews that they would be anonymized.

4 Empirical Findings

The empirical findings from the interviews are presented below. The findings are categorized with respect to the research model (see Table 2: Themes of the research model). The sub-themes are further categorized to present the findings in a concise and structured manner.

4.1 The Agile Method

Some details related to the projects the interviewees work with, their project teams and common risks they face are presented below. Thereafter their agile process is analysed to find risk mitigating aspects within their agile process.

4.1.1 Risk Mitigation with Agile Methods

The interviewees were asked about the general agile process they follow to analyze what aspects of it may have an impact on how risks are managed in their projects. Aspects of the interviewees' agile process which mitigates risks are given below. Amongst the interviewees, the risk management process described by I5 seemed to be the one most dependent on their agile method.

According to I5's description of their risk management process, risk identification, assessment, planning and monitoring exists when you work in an agile manner. All work is kept transparent between the team and the customer by having access to a prioritized product backlog. Sprint planning and grooming is done to plan future sprints. Working with daily standups allows one to plan ahead and improve transparency. Sprint review and retrospective is used to analyse the value which has been created by the sprint. Velocity can be used to estimate how much the sprint will be done with (I5, row 51).

While this process summarizes how agile methods can mitigate risks, further details and aspects are described below.

Agile techniques:

All of the interviewees follow an agile process mainly based on Scrum, with some traces of Kanban or Lean. However, none of them follow Scrum completely or have a defined agile process which is identical in every project.

I1 and I3 state that the main things they follow in every project are the agile values and principles. According to I1, when the project team has a certain level of maturity and experience with working in an agile manner, there is no need to have a clearly defined agile process to follow (I1, row 9). I5 points out that having the same agile mind-set in every project, even though the process may differ, helps the project team members to switch

between projects (I5, row 16). Similarly, I3 points out that the project team decides how they implement the agile mind-set and what tools to use (I3, row 10).

I1 believes that a big risk with software projects is to build the wrong things and the waste generated by rework. Agile processes prevent this type of risk effectively. Aspects such as *prototyping*, *due diligence* are used as a form of risk management before large investments are made (I1, row 13). I5 also states that agile methods *encourages change* while developing instead of fixing bugs at the end of the project (I5, row 30).

I2's projects tend to have three environments (Testing, Staging, and Production) which the solution is deployed to. These environments are used to *test* the solution thoroughly before it goes live (I2, row 17).

I5 mentions how their sprints, which are shorter than one month, allows them to build *just-in-time*. The current situation and context can be taken into consideration instead of building and *prioritizing* falsely. A *retrospective* phase at the end of each sprint is also used to evaluate the progress, changes and decisions (I2, row 3; I5, row 30). Daily *standups* have also proven to be effective for identifying problems and risks (I2, row 3).

I5 also states that *refactoring* can be done to mitigate future risks, even though it may not add any apparent value for the customer (I5, row 39).

Iterative and incremental development:

I1 and I5 mainly describe the iterative and incremental development process of agile methods as a risk mitigating aspect. I1 believes risks resemble a sinus curve, they go up and down and it is difficult to know which risks will come. Therefore, continuously making new plans eases the management of upcoming risks (I1, row 25). Agile methods with their iterative development process helps with this, since the project team raises the project risks as they become more apparent (I1, row 13).

I5 also mentions how new plans can be made as more is learned about the project as it progresses. Thus, the plans can be adjusted and unlike traditional projects, agile projects attempt to deliver the optimal value for the customer (I5, row 30). The risk of creating things which do not generate any value, in the form of return of investment, are mitigated thanks to iterative and incremental development (I5, row 31).

I5 states that the backlog should adjust to the context and priorities of the project. This is a result of working iteratively. When new requirements are added, other requirements may need to be removed or reprioritized with respect to what generates the most value (I5, row 35, 37). I5 points out that according to agile, everything should generate value. Incremental development assures that further value is added with every increment, partially by reprioritizing tasks (I5, row 42).

Transparency and customer collaboration:

According to I2 and I5, in order to effectively work with an agile process, the customer needs to be experienced.

I2 states that the customer often desired to carry out the project in an agile manner, but lack the experience or maturity to do so (I2, row 1). I5 states that a customer inexperienced with agile methods may not have an active product owner, which will have consequences on the schedule and collaboration with the customer (I5, row 20). However, all of the interviewees claim to collaborate closely with their customers, and there is a representative or customer owner in each of their projects.

I1 states that it is a requirement that the customer has a product owner, someone who takes all responsibility. The customer even sets up the project team with respect to their skills and requirements (I1, row 3, 5).

I3 and I5 also state that having a product owner available is vital since they know the business value. Choices and priorities need to be discussed with them (I3, row 35, 63; I5, row 32).

According to I1, the customer and the project team have the same goals, therefore everything within the project including the budget and priorities are transparent (I1, row 3). I5 believes that with agile methods, everything becomes transparent. Thus, preventing gold plating and prioritizing tasks becomes easier (I5, row 40).

I2 points out that risks or problems can be found during daily standups, where the customer also can be involved. Hence, with a high level of transparency, the customer can give quick feedback (I2, row 3).

Compared to traditional methods:

The interviewees were asked about the difference between risk management in traditional and agile methods to understand how the necessity and the practice of it differs.

I2 believes that risk management is more vital when working with traditional methods, but it is still necessary and unavoidable in agile projects (I2, row 41). I1, I3 and I5 point out that traditional methods attempt to predict every risk initially. Based on this, a plan is made for the entire project. Therefore, initial risk management becomes vital in traditional projects. However, a lot happens and changes during the course of the project. The team learns very much during the project (I1, row 41). Agile methods identify risks throughout the project. Thus, agile is able to adjust to the changing factors and have a greater chance to meet the latest customer needs (I3, row 59; I5, row 59).

I5 also points out that traditional methods met all the requirement set from the beginning and followed the project plan to the point, yet the customer was often displeased by the result. Agile methods avoid this by building just-in-time, focusing on the most viable product and not over-doing it (I5, row 31).

4.2 Agile Risk Management

4.2.1 Integrating Risk Management with Agile Methods

The interviewees answered questions related to whether and how they have integrated aspects of risk management with their agile process. According to the interviewees, none of their companies have an ARM process which is used by every project manager in every project. I1 states that the project team decides how to manage project risks, they do not use any clearly defined risk management process (I1, row 15). I2 claims that large organizations may have a fixed risk management process for every project, but not in I2's organization (I2, row 5). I4 also states that they have no consensus on how every project manager should manage risks, it is up to them (I4, row 1).

I3 and I5 point out that there is always some aspect of risk management implemented in their agile projects, even though their ARM process is different in each project, just like their agile process. Working with agile includes taking risks into consideration (I3, row 24; I5, row 53, 55).

Additionally, I3 states that sometimes aspects of traditional risk management can also be implemented (I3, row 24). I3 believes that risk management should be used in agile project, but agile tools and mind-set can also be utilized. Adding risk items to the backlog if it adds any value for example. It should not be regarded as a separate activity, it needs to be integrated naturally (I3, row 41).

Although it may be different in every project, the interviewees mention some aspects of risk management which are implemented in most cases. I1 states that risks are treated like any other item in the backlog, and can be managed by time boxing them, saving a specific amount of time to analyse and resolve the risks (I1, row 19). I3 also states that risks are added to a risk adjusted backlog, then compared and prioritized with respect to other items in the backlog and their business value (I3, row 26). The identified risks are either added to the risk register, or to an activity list or assigned to the team whom becomes responsible for them (I4, row 29).

Further commonalities in the interviewees' risk management are categorized and presented below.

Pre-project risk management:

I1 states that there are organizational risks, and technical project risks (I1, row 11).

Organizational risk can be analysed initially before the project starts to do a risk analysis and evaluate what challenges will be faced (I1, row 13).

I2 always uses risk management or risk planning before the project begins. I2 must know what external factors may affect the project to plan a schedule for the project. Hence, risks are calculated and assessed to analyse how the external factors may affect the project. To identify controllable risks and which actions need to be taken if they become a threat (I2, row 1).

According to I5, certain risks could be managed before the project starts, whereas others are managed throughout the project. Choosing amongst third party suppliers, thoroughly planning user stories may be a part of pre-project risk management while meeting the deadline, scope, budget are risks, which are managed throughout the project (I5, row 34, 42, 43).

Traditional risk management phases:

The identification phase is often included by the interviewees one way or another. While I1 mainly mentions an identification phase, I2 and I3 give examples of how the assessment and planning phases also are included.

I1 states that risk can be taken into account before the project begins, together with backlog grooming or in an estimation meeting (I1, row 33).

Similarly, I2 and I3 commences the project with a workshop where risks related to the project (such as the environment, development, customer etc.) are identified with some brainstorming. Thereafter they are prioritized with respect to their exposure levels (I2, row 11; I3, row 26). I5 also identifies risks in a higher level at the beginning of a project (I5, row 30, 34).

I2 states that risks with an exposure level 1 must be planned immediately (I2, row 11). I3 occasionally includes the customers in risk workshops (I3, row 26). I3 also tends to host QA workshops where risks are considered while analysing various risks, similar to a sprint planning session. Thus the risk are identified and appropriate actions are planned (I3, row 51).

I3 also mentions that risks are occasionally monitored throughout the project to assess how they have been managed, but it is something they strive to improve (I3, row 26).

Risk ownership:

While most of the risks are managed by the project team, not all of the interviewees assign risks to specific individuals.

While the project team manages project risks, organizational risks should be managed by the project manager (I1, row 11).

I2 mentions that the project managers or the customers who are responsible for the project are also responsible for the risk plan (I2, row 13). I3 states that it is the project manager's responsibility to ensure that the risks are managed, by planning for risk workshops or monitoring the risks for example. The risks need to be prioritized by the product owner and customer (I3, row 31, 33, 35). I5 repeats that the project manager owns the risks. However, it is the project team's responsibility to take risks into account with the work they do (I5, row 39).

In I2's risk management process, an owner is given to a risk after it has been identified and assessed. Thus the owner becomes responsible of keeping track of and managing the risk (I2, row 11). Similarly, I4 states that every risk is assigned a risk owner, otherwise the risk loses traceability and no action is taken against it. If the risk grows, it can be collaboratively owned

and managed by other developers or managers (I4, row 17, 19). I1 further repeats that the project team is responsible and take ownership of project risks (I1, row 13).

Common risks:

The interviewees were asked which risks they face most frequently in order to get an understanding of how well they may be mitigated by agile methods.

In I1's experience, stakeholder involvement and communication within the project team are the main risks in agile projects (I1, row 11). I3 and I5 also mention customer collaboration as a common risk. While I3 states that having a product owner mitigates this risk, I5 believes that finding an active product owner is challenging (I3, row 63; I5, row 20, 22). I5 further argues that only the customer knows what business value the system has for the customer. Not generating value for the customer is a major risk, and the product owner is a key element in preventing this (I5, row 22).

I2 mentions being dependent on third party suppliers, scope creep and testing as common risks. However, it depends very much on the project complexity and the customer (I2, row 7). Similarly, I4 finds it difficult to find a risk which comes up in every project, while mentioning infrastructure and integration as problematic areas (I4, row 14).

I2 believes experience helps with managing common risks, although there are no ready solutions for them (I2, row 9). Additionally, I5 states that the common risks with IT projects used to centralize around not finishing the project in time or budget. In agile, it is more important that the project delivers the desired value to the customer (I5, row 49).

4.2.2 Use of Artefacts

The use of artefacts may imply having a certain risk management process. All of the interviewees beside I5 state that a form of risk register is often used, while other artefacts are more situational.

Both I1 and I3 utilize the backlog also as a form of risk register. Risks are identified and treated the same way as any other task or work item in the backlog (I1, row 17, 19; I3, row 29, 53). I2 and I4 instead mainly use a simple risk register to keep track of identified risks (I2, row 33; I4, row 14).

Other artefacts are not regularly used by the interviewees. I1 states that risk boards, risk burndown charts have been used in some projects (I1, row 13). Similarly, I3 mentions that risk burndown charts have been used when risks which are manageable by the project team were needed to be visualized (I3, row 26, 27). However, there are no specific artefacts that are used all the time. The project team experiments with and tries to find suiting artefacts (I3, row 37).

I5 also states that the use of artefacts depends on the situation, if there are only a few risks, artefacts may not be essential to highlight them, they are not used in every project (I5, row 45).

I1 suggests that a good way to work with ARM is to have a risk register to raise risks and problems in the daily standups. Continuously looking at risks, assessing and reassessing them (I1, row 47).

4.3 The Necessity of ARM

4.3.1 Effects of ARM

Benefits:

One benefit which both I2 and I4 raises is keeping track of risks in large projects. I2 mentions that risks which come up during a large project are difficult to keep track of without risk management. Neither is it possible to predict all of them from project initiation (I2, row 1). I4 notes that all risks cannot be managed in a single iteration. They would need to be monitored throughout the project and handled in time. When there are many variables and the project has a long schedule, risk management becomes essential (I4, row 25).

Another benefit I4 mentions is the ability to focus on the daily work without constantly having to think about risks. The prioritized tasks in the backlog are taken care of instead while the risks are monitored (I4, row 33).

I2 believes that risk management also helps with understanding the customers' requirements. While risks are identified during the initial brainstorming sessions, the requirements are also better comprehended. The requirements can hence be discussed with the customer about interpretations in order to avoid misunderstandings and building the wrong things. Risk management helps with thinking in different areas, and may help with avoiding such misunderstandings (I2, row 33). I3 also mentions that without risk management, things may be built in wrongly and have flaws (I3, row 55).

In I2's opinion, having a standard process for risk management in agile projects would harmonize risk management throughout the company, thus make it easier for project managers (I2, row 19). Especially if the customer or the project manager is new. The process however should be flexible and adapt to the complexity and size of the project (I2, row 43).

I3 believes that risk management phases such as risk workshop can be integrated with agile processes and generate value. This would not contradict with the agile values or principles if done effectively. However, the value generated by the risk management phase must be compared with other tasks in the backlog. If it does not generate any value, the project team may not prioritize it over other tasks (I3, row 39, 45). Similar to I3, I1 also believes the use of risk management phases depends on the value it generates.

I1 mentions that all forms of risk management helps with easing the completion of a project and avoiding pitfalls, but to what extent it should be implemented is very project or customer specific (I1, row 35).

Consequences:

While I2 and I4 states that ARM would not have any consequences on the agile process (I2, row 33; I4, row 31), I1 and I3 have different opinions.

I1 implies that there is no risk management process that fits all projects. Having such a process would generate waste (I1, row 27). Since it would slow down the development as the project team will be following a process just for the sake of following a process. This would not generate any value (I1, row 31).

A consequence which I3 raises is that having too many side-tasks beside the main tasks may be against the “individuals and interactions over processes and tools” value stated in the agile manifesto. I3 thus implies that having a separate risk management process may contradict with the agile mind-set. It would have to be appropriately integrated with the agile methods and not be an external process (I3, row 45, 47).

4.3.2 Factors Affecting the Necessity of ARM

I1 and I4 point out that it is difficult to say when ARM would or would not be necessary in agile projects. It depends on the situation of the project (I1, row 29; I4, row 1).

Certain projects are more prone to risks than others, and such projects would require further risk management phases than a project which is progressing smoothly (I1, row 13, 29).

I2 and I4 on the other hand state that risks are unavoidable, and projects may face unforeseen situations. Having risk management would help both the customer and the project team (I2, row 21; I4, row 41).

Project size and complexity:

To get a better understanding of their agile projects, the interviewees were asked what they work with and how many are involved in the projects.

	Team size (people)	Type of Project	Reference (row)
I1	Small	Custom solutions.	Corrupted, 47
I2	3 to 8	Custom solutions. Intranet, Extranet, E-commerce.	1, 12, 29
I3	2 to 10	Custom solutions. Everything from mobile apps to complex integrated systems.	12, 14, 16
I4	2 to 6	Intranet. Mainly their own product. Also custom Intranet solutions.	3, 5
I5	1 to 5 Largest project has 14	Custom solutions. CRM, BI.	24, 26, 28

Table 4: Summary of project details

All of the interviewees state that they work with custom solutions of both large and small scale, complex and simple projects.

I2 points out that being more than 8 people in a project may make the communication within and management of the team difficult (I2, row 29). I5 states that in their organization, the team size may grow and shrink, but everyone works in the same location to ease communication and daily standups (I5, row 26, 28).

I1 and I3 states that they occasionally work with distributed teams (I1, row 7; I3, row 16).

I1, I2 and I4 were unison on believing that the size and complexity of the project affected the necessity of integrating risk management with agile methods.

I1 suggests that certain projects are less prone to risks than others (I1, row 13). Small projects without risks do not need an ARM process (I1, row 31). Two experienced developers working side-by-side most likely do not need a risk management process which will cost time. They can manage risks on their own instead (I1, row 35). However, in a large company where there are 150 developers, such as in a SAFe (Scaled Agile Framework) project, risk management processes may be essential since risks are a natural part of such large projects (I1, row 43).

Similarly, I2 states that in simpler, smaller projects such as external web projects, risk management may not be necessary since there aren't many factors affecting the project, hence it is less prone to risks (I2, row 15, 23).

I4 mentions two types of projects. If they are delivering their own product, it is done within weeks and there is not much development required, mostly integrating the product to the customer's environment (I4, row 3). If it is a development project for a custom intranet, the team size may vary between two and six people (I4, row 5). I4 also suggests that short projects do not require extensive risk management, but the communication needs to be flawless and there should not be many variables affecting the project. I4 states that such projects are rare, and having some sort of risk management would be beneficial to be aware of possible risks (I4, row 27). Similarly, I1 and I3 states that regardless of the type or team size of a project, risks need to be taken into consideration (I1, row 27; I3, row 49, 51).

Team maturity:

The maturity and experience of the project team was raised as a factor by I1, I2 and I4.

I1 states that the more the project team knows, the more variety of views they can have on risks. Even if the project team misses a risk initially, they are likely to catch it as the project progresses (I1, row 33). I1 also states that agile demands discipline and maturity, which not all organizations have. The absence of maturity may increase the need for a clearly defined risk management process to support their project teams (I1, row 43).

I2 confirms this by stating that working with an experienced team who has been working together over an extensive period with a certain customer, with good team dynamics, is able to work well with agile and not require a defined risk management process (I2, row 27, 41).

I4 points out that the risks which arise often have straight forward solutions. They may cause delays, but the team often know how to handle them (I4, row 23). Similar to I1 and I2, I4 believes that if the project team has done something many times and knows exactly how to

carry out the tasks and what to expect, a risk management process may not be necessary. Though this is most unlikely according to I4 since the customer or the project may face unforeseen situations (I4, row 41).

Customers:

Certain customers have their own risk management process due to legal or industry specific reasons. I1 gives a medical company as an example, where they directly affect the lives of people. Such companies may have their own risk management processes. In such circumstances, I1 and I2 states that the customer's risk management process must be used (I1, 29, 45; I2, row 15). Both I1 and I2 states that the bigger the organization of the customer is, the more likely it is that they will demand a risk management process to be followed (I1, row 43; I2, row 15).

I2 states that acclimatizing to the customer's risk management process often is not a problem since the project team is familiar with them from their own risk management processes (I2, row 15). I1 on the other hand mentions that the customer's risk management process may not generate much value, but still must be followed (I1, row 29).

Generating value:

A reoccurring theme which I1, I3 and I5 raises is generating value. The main point with agile methods is to generate value for the customer.

I1 underlines that the main point is to question everything that is done in an agile project. If there is no value behind a task, then it should not be done. Likewise, if ARM does not generate any value, it is not necessary and a more lightweight system could be used (I1, row 47).

I3 states that risk management does not contradict with agile methods. However, the project teams experiment with various artefacts or risk management phases and only utilizes them if they generate value (I3, row 65, 67).

I5 on the other hand believes that risk management processes in addition to agile processes would not generate much value. Yet this would require excellent communication within the project team and a well-managed backlog with clear and detailed items (I5, row 47).

5 Discussion

5.1 The Agile Method

5.1.1 Risk Mitigation with Agile Methods

Although the interviewees base their agile methods on Scrum, they do not have a standard agile process that they follow. They are mainly following agile values and principles, in other words, the agile mind-set. In their agile projects, they claim to manage risks through working in an agile manner. Some interviewees, especially I5, believes the use of various agile techniques fulfils the same purpose as traditional risk management phases.

As stated by Alleman (2002) and Sommerville (2016), agile methods strive to maintain simplicity and escape the complexities of traditional methods. This may partially be accomplished through the use of agile techniques. They tend to be simple and short phases which can be implemented in the iterations or increments of a project. The interviewees mention their use of agile techniques such as having a prioritized backlog with backlog grooming, sprint planning, daily standups, velocity calculations, retrospective sessions, refactoring etc. All of these agile techniques in one way or another have a risk mitigating influence on the project. A retrospective session can be compared to a post-mortem risk management analysis phase. Standups can be used for risk identification. Code refactoring reduces future risks by reducing complexity. Demos and prototyping is also used as a manner of ensuring the customer knows and receives what they invest in. Change is encouraged and embraced, rather than fixing bugs late in the process. Although these techniques are not exclusive to agile methods, they are building blocks for them.

Both the interviewees and the literature review imply that the iterative and incremental development strategy of agile methods is the central risk mitigating aspect. I1 compares risks to a sinus curve that goes up and down throughout the project. New plans are made to adapt to the changing circumstances and risks that become more apparent as the project progresses. New requirements or identified risks are taken into consideration with every new plan, which eases the management of risks and strives to deliver the optimal value to the customer as well. Hence, the backlog is updated with the most prioritized tasks. As Coram and Bohner (2005) puts it, an evolving list of requirements thanks to iterative and incremental development helps with handling shifting requirements, which is where risks often arise. Thus, changes and risks are responded to quickly by embracing change (Henninger et al., 2002).

Another essential risk mitigating factor of agile methods is the close collaboration with the customer and transparency in the project. Some interviewees mention that the customer must be experienced or knowledgeable within agile methods to have an effective collaboration with the project team. While this is raised as a concern or common risk by the interviewees, they all underline how crucial customer collaboration is. Only the customer fully knows the

business value of the project. The project must be shaped by discussing priorities and choices with the customer. This is eased by striving to keep things such as the backlog, budget, and priorities transparent between the customer and the project team. As Lindvall et al., (2002) and Dingsøy et al., (2012) state, the central idea is to deliver the customer what they invest in. To do so, there must be a constant contact between the customer and the project team, where the customer can demand changes in time and the customer is consulted regarding priorities. By doing so, risks such as gold plating or scope creep are mitigated while striving to generate the most value for the customer and reach project success. Standups where the customer also is involved are shown to be an effective technique to achieve this.

Comparing traditional methods with agile methods, risks are mitigated to a further extent in agile methods. Traditional methods required explicit risk management phases initially, and had to foresee possible risks to handle them. In agile methods, risks are identified as the project progresses. Once identified, agile methods can adapt to the changes. Traditional methods were not known to adapt well to changes which came during the project due to following a sequential order of software development (Sommerville, 2016). Therefore, both the literature review and the interviewees believe agile methods require less explicit risk management phases compared to traditional methods. As I5 puts it, meeting the initial requirements of the project, as traditional methods do, does not always lead to project success. Generating the most value in terms of return of investment does, which is what agile methods are effective in doing.

5.2 Agile Risk Management

5.2.1 Integrating Risk Management with Agile Methods

Every project is different and ARM needs to be customized to suit the project at hand (Moran, 2014). The interviewees are of the same opinion, that none of their projects are the same. They do not have a specific agile process that they follow, nor do they have an explicit ARM process. The project team and project manager decides how to manage a project and its risks. They state that large organizations may need an explicit process, but not the interviewees' organizations. However, there is always some aspect of risk management implemented in their agile projects. I5 points out that working in an agile manner includes managing risks. Aspects of both traditional risk management and agile techniques can be implemented as long as they are consistent with the agile mind-set (Andrat & Jaswal, 2015).

Traces of the pre-project management phases described by Nyfjord (2008) and Moran (2014) regarding *project context and risk environment* analysis and *risk scoping* can be seen in some of the interviewees' descriptions of their projects. Two interviewees differentiate organizational risks that are managed before the project begins, from project risks, which are managed throughout the project. Another interviewee mentions how risks are assessed prior to the project to make decisions affecting the whole project.

In this way, although they do not have a defined process for the pre-project management phases or other risk management activities, the interviewees appear to carry out steps similar to an ARM process. On the other hand, the interviewees give examples such as choosing amongst third party suppliers, analysing what external factors may threaten the project or allocating necessary resources and individuals to the project. This may be interpreted as a natural step for software development projects. While such steps do have an impact on the risks of the project, they are likely to be evident steps and not carried out solely for the sake of pre-project risk management. This can be seen as an indication of how risk mitigation and management can be implicitly involved in the agile methods.

When it comes to traditional risk management, the phase that is mostly adopted by the interviewees is the identification phase, together with a risk register. The interviewees use different methods such as estimation meetings, backlog grooming or QA sessions and risk workshops with brainstorming to identify risks. Two interviewees also mention that they assess identified risks by analysing their exposure level, or that they monitor risk. Once the risks have been identified, assessing, planning and monitoring them becomes a streamlined process. They can be carried out if a certain risk poses a threat to the project. This may be the reason why the interviewees mainly mention their risk identification phase, but not the others.

For a risk to be managed, an owner ought to be assigned to it. As I4 states, without a risk owner, no action would be taken against a risk. How risks are assigned to an owner is somewhat different both in the literature review and amongst the interviewees. While Moran (2014) and three interviewees believe a risk should be treated like an ordinary backlog item where someone will eventually be assigned to it, Nyfjord (2008) and two of the interviewees prefer to assign a certain individual or group to the risk as soon as it is identified. Despite these differences, risks eventually always receive an owner. Thus, the risks are not ignored or forgotten. While the project manager may technically own the project and risk plan, the project team must manage them.

The literature review had contrasting views on whether agile methods mitigate risks enough to not require an explicit ARM process. When the interviewees were asked about the most common risks they face in their projects, the most mentioned risk is customer or stakeholder collaboration. This may be a risk which arises more often in agile methods since customer collaboration is an essential, yet demanding aspect of agile methods. The customer must have an active product owner whom can participate in and guide the project towards the desired outcome. Some of the other risks mentioned by the interviewees such as scope creep and not generating value is mitigated by the agile principles (McConnell, 1996; Albardaneh et al., 2015), while some such as third party suppliers, integration of the product to the customer's environment and testing may not be mitigated by the transition from traditional to agile methods. This implies that agile methods have their own risks, and do not mitigate every risk simply by following the agile principles. This is where the argument for the necessity of ARM arises, to manage risks which are not directly mitigated by agile methods.

5.2.2 Use of Artefacts

The use of artefacts can be seen as some form of integration of risk management with agile methods. Artefacts for risk management are tools that are used for visualizing the circumstances and communicate the risks effectively to the project team. They make risks more transparent and the project team more aware of them. Risks can be identified, assessed, planned or monitored through the use of artefacts. In this way, proactive or reactive measures can be taken towards risks (Moran, 2014).

The interviewees do not utilize many artefacts except for some form of a risk register. With a risk register, new risks are identified and added to the list of risks together with details about them. The existing risks are also updated. It is mainly used to identify and understand existing risks in order to later assess, plan or monitor them (PMBOK, 2016). Moran (2014) suggests that a risk register should be kept and updated throughout the project, since new risks are identified through changing iterations and increments.

Some of the interviewees use the backlog as a risk register, which can be an effective way to naturally integrate a risk management artefact with agile methods. Treating a risk like a backlog item and associating project team members to it. This is in line with the agile process while it also fulfils the same purpose as a risk register. They may use it to raise risks and problems in daily standups. They can continuously look at risks, assess and reassess them. As Il says, agile is all about planning. By being aware of the risks, they can be assessed, planned and monitored if needed. The project team can take the current context into consideration while planning and shaping an iteration or increment. Thus, the identified risks may be handled.

Other artefacts such as risk boards and risk burndown charts have been used in some projects, but they are situational while a form of risk register seems to be used in almost every project. What artefacts are to be used in a project is decided by the project team. They experiment with various artefacts, and know what to use in specific situations. If the project is not prone to many risks, artefacts that visualize them may not be necessary.

5.3 The Necessity of ARM

5.3.1 Effects of Risk Management

As it was in in the literature review, the interviewees mentioned more benefits with risk management than consequences.

Risk management helps with handling project uncertainties by becoming aware of, understanding and communicating risks (Moran, 2014). Risks may also affect other critical success factors for the project. They need to be prioritized and effectively managed (Boehm, 1991; Sommerville, 2016). Therefore, risk management is regarded as a critical success factor (PMI, 2009). Practicing it effectively increases the chances of project success (Moran, 2014;

Gold & Vassell, 2015). ARM is also found to increase stakeholder confidence and contribute to the estimation, planning and design phases of the project (Chin, 2004; PMI, 2009).

The interviewees suggest that ARM allows the project team to focus on the tasks at hand rather than pondering on risks. It helps with keeping track of risks, especially in large and complex projects which have long delivery times. The customer requirements are also better understood through risk identification. They can be discussed with the customer and misinterpretations can be reduced. I2 even believes that having a flexible guideline for ARM may harmonize the risk management process throughout the projects, and make it easier for project managers. As I1 states, all forms of risk management eases the completion of a project and avoiding pitfalls.

When it comes to consequences that may arise with the implementation ARM, Nyfjord and Kajko-Mattsson (2007) argue that ARM may be against the agile principles and values and may not be optimal for small projects. Disrupting these values and principles would only cause inconvenience rather than be beneficial. While two of the interviewees believe that ARM does not have any consequences on their agile process, two other interviewees argue that if it is not generating any value, it is not necessary. It would contradict with “individuals and interactions over processes and tools” value, and generate waste if a process is followed only for the sake of following a process.

Implementing ARM falsely may lead to these consequences. However, as Moran (2014) and Ylimannela (2011) stated, ARM must be implemented in accordance with the agile values and principles. It should be integrated seamlessly with the agile process. In other words, having an explicit ARM process, which is used as a side-process, will be cumbersome. However, implementing valuable aspects of risk management, such as identifying, assessing and planning for risks and adding them to the backlog as a work item may prove to be beneficial without contradicting with the regular flow of the agile process.

The interviewees are aware of the benefits of risk management, and list some of them which are not mentioned in the literature review. Whether all of these benefits are a direct result of ARM is debatable, but the interviewees deem ARM to be a supportive factor in these benefits. Despite these benefits, they do not have a fixed ARM process. This may suggest that the same benefits can be acquired without an explicit ARM process, and simply applying the required, most value generating aspects of risk management with respect to the project.

5.3.2 Factors Affecting the Necessity of ARM

Both the interviewees and the literature review implies that every project has risks. The larger and more complex the project is, the more likely it is that there will be more risks, and ARM will be necessary (Moran, 2014). Three of the interviewees also raised project size and complexity as factors affecting the necessity of ARM. They point out that small and short projects which do not have many external factors of variables affecting the project may not need extensive ARM. However, they mention that communication within the team must be flawless in such circumstances.

However, as I1 points out, to what extent ARM should be implemented is very project or customer specific. The value generated by risk management phases must be compared with other tasks in the backlog. If a risk workshop for example does not generate more value than another task, it may not be necessary or prioritized.

On the other hand three of the interviewees mention that risks must be taken into consideration regardless of the type or size of the project. Although certain projects are less prone to risks than others, risk are unavoidable and a project may face unforeseen situations. Although there is no boundary drawn between a large and a small, or a simple and a complex project, I1 mentions that in comparison to SAFe projects, their projects are relatively of smaller scale. The necessity of ARM may be much more apparent in a project which uses a framework such as SAFe with a project team comprising of dozens of individuals. However, risks seem to be managed more implicitly in the custom solution projects I1 and most likely the other interviewees work with.

Another factor mainly raised by the interviewees was the maturity and experience within the project team. An experienced team which has been working together with the same customer and technical environment may not need ARM. A mature project team may find straight forward solutions to problems or risks which arise during the project. The interviewees also point out that the project team learns as the project progresses. The more they know, the more variety of views they may have on risks. Given the iterative development in agile projects, a better understanding of the whole project is achieved through the iterations, and new the project team may have new and more accurate perspectives. However, relying on the project team rather than ARM to handle risks demands mature and skilled project team members. Agile methods rely on skilled individuals who take on responsibility (Sommerville, 2016). I1 also states that agile demands discipline and experience, which not all organizations may have. In such organizations, ARM may be more of a necessity as a support for the project teams.

The most frequently mentioned factor in the interviews is the aspect of generating value. Agile methods strive to generate value for the customer. ARM should only be implemented if it generates more value in comparison to other tasks. I5 believes that an explicit ARM process would not generate much value since most of the risk management and mitigation is done in the agile process. As long as the team has good communication and a well-managed backlog, agile methods may suffice with its implicit risk management.

A factor which was not noticed in the literature review yet brought up by two interviewees was how the customer may affect the risks management process. Certain customers have their own risk management process due to legal or industry specific reasons, where the project team must follow the customers' risk management process. While the project team may occasionally find these risk management phases to be rigorous, they must be followed.

Contrarily, in addition to project size and complexity, the literature review argued that distributed teams and the attitude towards risks also were factors affecting the necessity of ARM (Nyfjord, 2008; Moran, 2014). While these factors were not implied to be significant for ARM in the interviews, two of the interviewees mention that they work with distributed

teams. Although distributed teams may make communication more difficult, it allows the addition of skilled members to the project team.

6 Conclusion

Which aspects of risk management are implemented in agile projects?

Agile methods with their agile techniques, iterative and incremental development and customer collaboration are effective with delivering to the customer what they require. Most of the risks that arise throughout the project are implicitly managed by these aspects of agile methods, which also indirectly serves a risk managing function.

Although agile methods mitigate risks extensively, certain aspects and phases of risk management can be integrated with agile methods without sacrificing or contradicting the agile values and principles. While there is no solution that is valid for every project, an effective way of implementing ARM according to the findings of this study is to treat project risks like any other work item in the backlog. For this, risks must be identified before adding them to the backlog which acts as a risk register. The risks can hence be assessed, planned and monitored by the project team. Members of the project team can take responsibility for them when actions must be taken against them. Taking basic project risks into consideration while assessing and making initial plans about the project also helps with managing such risks. Thus, certain risks can be avoided during the project.

Creating an ARM process in this way increases the odds of project success while also facilitating the certain aspects of the project, such as focusing on the tasks at hand, planning and estimating or avoiding misinterpretations or pitfalls. Despite these potential benefits of ARM, they must be weighed with respect to the project and against other tasks or work items in terms of what generates the most value for the customer. Following an explicit ARM process without any valid cause would be wasteful, which is what agile methods strive to avoid.

The main factors that are found to play a role in the necessity of ARM is the size and complexity of the project as well as the maturity and experience of the project team. The larger and more complex the project is, the more likely it becomes that ARM will have greater contributions to the project and will become necessary. The necessity of ARM is also believed to increase when the project team is inexperienced and could use a process to increase the chances of project success by managing risks affecting the critical success factors of the project.

7 Appendices

7.1 Interview Request Template

The interviewees were Swedes; therefore, the interviews were conducted in Swedish in hopes of making it easier for them to respond to the questions naturally and comprehensively. Before each interview, the research problem and the study in general was explained to them both via email and in person.

The explanation given to the interviewees through email is as follows in Swedish:

Enligt litteraturen är en projekt risk, definierad som en oförutsebar händelse som har en positiv eller negativ inverkan på projektet eller organisationen. Man har hanterat mjukvaruprojektrisker på olika sätt, allt för att öka chanserna av att leverera det som kunden vill ha, således öka chansen av framgång. Det började med att använda sig av metoder för att driva projekt. Traditionella eller plan-baserade mjukvaruutveckling. Genom att följa en viss standard var man mer benägen att nöja kunden. Kunden uppskattade en sådan utvecklingsprocess. Dock var detta inte tillräckligt, det dök upp risker i projektet. Därför implementerade man riskhanteringsprocesser, som identifierade, analyserade, övervakade och bemötte projektriskerna. Därefter kom agila metoder, och dessa metoder mitigerade risker som var allmänt kända med traditionella metoder.

Idag finns det två infallsvinklar på riskhantering när det gäller agila projekt. Den ena vinkeln säger att man inte behöver riskhanteringsprocesser i och med agila metoder är en risk mitigeringsprocess i sig själv. Den andra vinkeln pekar på att agila metoder inte hanterar risker tillräckligt, och att en explicit riskhanteringsprocess skulle vara nödvändig. Medan vissa anser att detta skulle förbättra agila metoder, andra anser att det skulle strida emot agila principerna, därmed ha en negativ påverkan.

Det jag vill undersöka är hur företag tar ställning till detta. Hur man hanterar risker i agila projekt. Varför man har valt att hantera risker på det viset, vilka faktorer som spelar roll i det valet. Vilka effekter detta kan ha på projektet.

7.2 Interview Questions

- 1) Vad är din roll i företaget?
 - a. Hur länge har du haft denna roll?
- 2) Vilka typer av projekt har du jobbat med eller fortsätter jobba med? Kan du beskriva lite de projekt du har jobbat med?
 - a. Vilken storlek (komplexitet) av projekt arbetar företaget oftast med?
 - b. Hur många brukar jobba på ett projekt?
 - c. Jobbar alla på samma ställe, på kontoret, eller är de på olika geografiska ställen?
- 3) Använder ni (företaget) er av agila metoder? Vilka?
 - a. Varför valde ni att jobba agilt?
 - b. Följer ni agila metoder till hundra procent, eller gör ni era egna variationer av dem?
 - c. Vad har ni ändrat eller lagt till agila metoderna? Varför?
 - d. Varierar det från projekt till projekt, eller följer ni en allmän agil process på alla projekt?
- 4) Vilka typer av risker brukar ni identifiera med era projekt? Vilka är mest förekommande? (risker med kunden, projekt team, testning osv)
 - a. Vilken attityd mot risk har ni? Vilken grad av risk accepterar ni? Är det gemensamt för alla? (benägenhet, tolerans, tröskel)
- 5) Använder ni er av "Risk Management", eller risk hantering i era agila projekt?
 - a. Om ni har risk management;
 - i. Hur ser det ut i detalj (identification, assessment, planning, monitoring)?
 - ii. Hur har ni integrerat det med att jobba agilt?
 - iii. Vem eller vilka tar ansvar för risker med ett projekt? Vilka är inblandade med riskhanteringen?
 - iv. Skiljer sig riskhantering från projekt till projekt? På vilket sätt?
 - b. Om ni inte har risk management;
 - i. Upplever ni att ni era projekt är riskfria?
 - ii. (Om de säger att de inte har haft risker, fråga om de har haft problem med att hålla budgeten, leverera i tid, att kunden inte kunde medverka, projekt teamet inte kunde prestera osv – olika scenarion)
 - iii. Identifierar ni risker under projektets gång?
 - iv. Tar ni åtgärder emot de identifierade riskerna? På vilket sätt?
 - v. Hjälper agila arbetssättet med att hantera risker? Hur?
- 6) Använder ni er av verktyg eller tekniker (artefakter) som risk burndown chart, risk register, risk modified backlog, risk matris eller liknande (för att visualisera situationen eller övervaka det)? Vilka, och på vilket sätt?

- 7) Hur upplever du er nuvarande riskhantering är? (Hjälpsam, redundant, bristfällig)?
 - a. Anser du att riskhantering är nödvändigt när man jobbar agilt? Varför?
 - b. Påverkar riskhanteringsprocessen era agila arbetsmetoder? Blir ni mindre agila, eller är det stärkande att implementera riskhantering? (dokumentation av risker, planera för riskhantering, ta tid från att utveckla till att hantera risker)
- 8) Finns det projekt där ni inte använder er av riskhantering?
 - a. Under vilka omständigheter skulle riskhantering inte behövas?
 - b. Vilka faktorer påverkar detta?
- 9) Anser du att riskhantering har påverkat framgång eller misslyckanden av era projekt?
Till vilken utsträckning? (för- och nackdelar)
- 10) Har du varit del av något projekt där traditionella, plan-baserade metoder användes?
 - a. Hur skiljer riskhantering i agila projekt från traditionella projekt?
 - b. Behöver agila metoder lika mycket riskhantering som traditionella projekt?
- 11) Tack! Finns det något mer du skulle vilja ta upp eller lägga till?

7.3 Interviews

Interview 1

Row	Speaker	Answer
1	D	Vad är din roll i företaget?
2	I1	Jag är regionansvarig, men jag har varit utvecklare och projektledare och har mycket god erfarenhet inom agile.
X	X	The recording was corrupted, and one minute of the interview was lost Continuing where the recording recovered...
3	I1	... inte kunna hamna liksom på efterkälke på grund av att systemet inte klarar av att integrera eller vad det nu kan vara. Så utmaningen är ju att lyckas få till den agila arkitekturen för det behöver man förstås liksom problemområdet och domänen och liksom vad det egentligen ska lösa i stor utsträckning. Och det använder vi agila principer till, för att ta reda på. Vi jobbar extremt transparent med våra kunder. Till och med så att dem är med och sätter ihop teamen utifrån vad de vet att dem ska ta vägen osv. Och vi kräver väldigt mycket så att säga kunddeltagande genom hela processerna och är väldigt transparenta alltifrån budget till hur det går till prioriteringar utan det gör vi ihop. Vi har ett gemensamt mål.
4	D	Får ni alltid det kundengagemangen, så mycket som ni vill ha?
5	I1	Levererar vi projekt här ifrån så är det ett krav. Annars så levererar vi inte projekten för det innebär att vi tar helhetsansvar för en hel leverans. Och då måste vi ha det committet från dem som vill ha det utvecklat så att säga. Ute hos våra kunder varierar de givetvis. Oftast - det är det som är skillnaden, att där är det kunden som är ansvarig för leveransen. Och vi är egentligen bara guns for hire som ska komma in.
6	D	Och när ni utvecklar era projekt, jobbar ni distribuerad på olika kontor?
7	I1	Vi har jobbat distribuerat också. Vi har flera kontor runt om i Sverige, och vi har även jobbat med kunder som sitter i Tyskland och levererat från två tre kontor. Det är en utmaning i sig. Att distribuera team är svårt.
8	D	Du nämnde att ni jobbar agilt och lean baserad. Men vilka variationer har ni gjort? Varför har ni valt att göra variationer istället för att följa Lean till 100 %?
9	I1	Jag tror att all form av metodologi är väldigt bra med stödhjul. Om en organisation behöver stödhjul. Har man en viss mogenhet inom agilitet så

		<p>slöar dem ner dig ungefär, och att du fokuserar mer på processen än vad du fokuserar på vad egentligen är grundproblematiken. Kanske du ser i början att du behöver väldigt mycket hjälp och guidning och tydliga riktlinjer liksom. Då kan det vara bra om man har processer, de kan vara bra för dig. Men ju mer mogen du blir och ju mer kunskap du besitter i varför du gör saker, alltså själva undermeningen med agile. Eller undermeningen med agile eller vad man än vill kalla det. Så börjar man se saker som att man behöver - det finns ingen religion att följa en process liksom. Det är inte huvudsyftet med den. Men det kan vara allt från hur vi använder root cause analysis för att liksom analysera om något går åt pipsvängen eller hur vi använder design sprints eller hur vi använder fokuset på slutanvändarnas upplevelse av ett system från Lean software development. Ur den kvalitetsaspekten. Perceptionen slutanvändarna har av ett system blir minst lika viktigt eller det blir viktigare än hur den interna arkitekturen ser ut till exempel. För att vi har olika principer inom bolaget som vi vill följa även när det gäller bolagsutveckling eller när vi levererar mjukvara. Och i det här förhållandet så för oss är det också, oavsett om det är bolaget eller ett projekt så är det liksom i en stor utsträckning tycker jag, det är just risk hantering.</p>
10	D	<p>Precis, det var en liten introduktion med hur ni jobbar agilt där. Men vidare till riskhantering, vilka typer av risker brukar ni identifiera i era projekt? Vilka är mest förekommande?</p>
11	I1	<p>Jag tror att det är generellt så är det stakeholder involvement eller liknande personer som behövs med i projektet men dem inte kan för att dem är chokeade med andra grejer att göra osv. Att de har flera roller ute i en organisation ute hos våra kunder. Och det är väl återigen så är det ju kommunikativa inom mjukvaruutveckling. Det är så otroligt viktigt. Det här klassiska, skit in, skit ut. Det stämmer. Så jag tror att det är en av de stora riskerna sedan finns det alltid - alltså det finns organisatoriska risker, sedan finns det kanske lite mer tekniska, projekt specifika risker. Och dem tror jag man får hantera på olika sätt. Jag tror att dem teknik, projekt specifika som mer än handlar om kanske constraint på något man ska bygga. Det kan vara att får vi den leveransen från den tredjepartsleverantören som tillhandahåller det API:et. Så den typen av risk liksom. Hela projektet, den hanteras på ett ganska bra sätt utav teamet. Medan organisatoriska problem är kanske då kanske någon agil projekt ledare eller Scrum master eller vad man nu vill kalla det som bör dra i. Och i ett moget agil team så kanske det inte behövs en Scrum master, utan då kan teamet kan, beroende på storlek, hantera det också. Det är också lite därför vi anammar oss inte någon specifik princip för att har man små projekt så blir kanske Scrum väldigt mycket ceremoni overhead. Och det skapar mer frustration och waste i processen. Och vi försöker minimera waste så gott vi kan.</p>

12	D	Du nämnde att ni kanske har en Scrum master eller agil projekt ledare. Men finns det några som är ansvariga för att hantera riskerna?
13	I1	Om det är vi som är leveransansvariga så är det vi och teamet som ansvarar för liksom prata om riskerna och försöka vara proaktiva åt dem. Är det organisatoriska risker så kanske man sätter dem initialt innan projektet och gör en liten riskanalys och ser vad det är egentligen som utmaningen är. I det här projektet, vad är det som kommer driva det här projektet, så att teamet vet hela tiden vad dem går in i när dem börjar och vad de ska jobba mot. Är det däremot att det är löpande projekt - alltså vissa projekt är mindre riskutsatta än andra. Liksom det beror lite på vad det är som ska lösas, vilken organisation det ska byggas i. Och i vissa projekt har vi kört sån risk registry, jag tror det var Boehm snackade om det först. Där man egentligen ser det som att man har en backlog med olika typer av risker som man sedan kan få en burndown chart där man också ser hur risk är mitigerad i projektet. För vissa projekt kan kräva lite tydligare - just att det är så liksom mitigerad, det kan vara både tekniskt och organisatoriskt. Och då får teamet jobba på det, även lyfta upp vad är dem potentiella riskerna. Sedan är det liksom den iterativa utvecklingen ganska riskhanterande per sig för att man liksom - på två veckor hinner man se om det går åt helvetet. Eller i alla fall var det är. Kan man göra åtgärder åt det? Och alltså den här stora risken i mjukvaruprojekt är ju alltså waster av rework, att man bygger fel sak. Om man kikar på det i en ren teknologiskt värde. Liksom risk och där har du dem agila processerna ganska mycket liksom inbyggt stöd för att hantera den typen av risk. Och vi jobbar mycket med prototyping och pretotyping inför ett projekt, innan det blir ett utvecklingsprojekt. Det finns för att hantera risk. Det kan vara till exempel att göra en teknisk due diligence på en typ av teknik som dem vill då använda, eller något liknande. Då gör vi en prototyp på det så att man inte investerar extremt mycket pengar initialt. Utan att man inser att - och det är den typen av risker vi kan kontrollera och hantera. Och så har vi valt att jobba.
14	D	Hur skulle du beskriva er riskhanterings process, från att identifiera risker till att övervaka och hantera dem?
15	I1	Jag tror att det varierar extremt mycket från projekt till projekt. Det är inte så att vi - om man kollar på SAFe och sådana, så kör dem roam och andra riskhanterings principer, men vi har ingen sån uttalad. Utan det är egentligen lite det som teamet som ska bygga det och dem som leder det.
16	D	Har ni integrerat risk hanteringsprocesser med agila processer på något vis?
17	I1	Det är i så fall om man kikar på typ att man har en risk registry eller så som vissa projekt har haft. Att man har en backlog på risker. Så att dem kommer upp till ytan. Det finns andra sätt att liksom - vi har haft risk board i något projekt där man egentligen bara lägger upp som en note, dem här är

		riskerna, så man kan stå och prata om dem och se hur dem betar sig. Om det är någon som blir mer kritiska eller så vidare.
18	D	Förutom den här risken registren och boarden, har ni andra artefakter eller verktyg som ni använder för att hantera eller övervaka riskerna?
19	I1	Det är väl egentligen - i en iteration så kanske - det är ju också ett sätt för agila processer att kunna riskhantera, det är ju att göra olika spikes, att man timeboxar något man vet - okej, detta är en potentiell risk som kommer om två månader. Vi måste kunna vara lite proaktiva mot den. Vi behöver 40 timmar för att undersöka denna risk. Och det hanteras som vilken annat krav som helst, för de är en del av projektet. Så det är också ett sätt som man använder under iterationens gång. Men det är främst relaterat till mer kanske tekniska risker eller till risker som går det här att byggas så som ni hade tänkt er. Kommer integrationen flyta, kommer prestandan vara tillräckligt bra osv.
20	D	Använder ni er av risk burndown chart eller liknande?
21	I1	Ja, vi testade det vid något projekt. Vi hade en sådan risk registry och såg hur det flyter liksom. Och det får ett viktad värde ungefär på hur stor är impacten och vad kommer det göra, hur stor är sannolikheten att den kommer inträffa.
22	D	Har ni några specifika lösningar till specifika risker? Till exempel om risk exposure är på denna nivå så ska vi ta denna åtgärd?
23	I1	De tror jag är lite upp till teamet. Det är inget som vi har definierat. Vi jobbar ändå med ganska hög risk projekt i form av kort, viktiga deadlines i form av time to market och liknande. Och ganska komplexa system med ganska mycket externa beröringspunkter både juridiska grejer som kanske kommer men även rent tekniska utmaningar.
24	D	Och vilken attityd har ni mot risk? Är det noll tolerans, eller har ni någon viss nivå ni kan hantera?
25	I1	Risk är lite som en sinuskurva, den går liksom upp och ner hela tiden i ett projekt. Vi vet aldrig vad som händer om sex månader. Det kanske är en leverantör som går i konkurs. Du kan inte planera för allt, det är just därför vi tror på iterativ utveckling. Och man brukar säga generellt om agile överlag men, många bolag säger nä vi slänger iväg Scrum eller något för vi tycker inte det funkar riktigt för oss, men så är de inte mogna. Så blir det någon form av agile eller väldigt ad-hoc medan vi är av liksom den erfarenheten att agile is all about planning and planning and planning and planning. Det är också ett sätt att hantera risk. Att hela tiden planera om, styra vad det är som är riskerna framöver och det sker ju precis som alla andra planering. Man kan ha roll in wave planering, där man kikar kanske

		på tre månader, då har man ganska bra koll på riskerna. Sex månader lite mindre. Ett år, ingen aning. Ja det kanske kommer ett nytt datadirektiv eller något liknande. Men att hela tiden omplanera även om i ett omtag på olika typer av risker. Det är därför vi vill jobba iterativt.
26	D	Anser ni att när man jobbar iterativt och agilt, måste man ändå ha en riskhanteringsprocess? Eller räcker det att jobba agilt för att hantera risker?
27	I1	Jag tror det beror helt på projektet. Återigen, det finns inget som funkar på alla. Jag tror har du en stor riskhanteringsprocess i alla projekt så kommer det skapa waste. Däremot så tror jag att pratar man inte om risk, lyfter inte upp det i dagordningen så har du inte fattat hur man utvecklar mjukvara. Men jag tror att vissa projekt absolut. Men jag vet också i många projekt som hade bara skapat liksom waste.
28	D	Och under vilka omständigheter skulle man inte behöva ha någon slags riskhanteringsprocess?
29	I1	Det är helt omöjligt att säga. Alltså vissa projekt är där jättemycket risker. Som jag sa, organisatoriska, tekniska risker - allt. Dem projekten tjänar förmodligen på att lyfta upp risk på ett lite tydligare sätt. För att det är så aktivt hela tiden om projektet. Dem behöver troligtvis någon typ av hantering för att kunna liksom lyfta upp riskerna mer på dagordningen medan andra projekt som kanske är lite mer smooth sails, liksom dem klarar sig ganska bra utanför och klarar dem agila liksom. Under meningen är att det är automatiskt. Sedan kan jag säga att det är - ute hos våra kunder där vi kommer in i deras verklighet, och vi jobbar mycket med säkerhetsbolag. Och de har sina definierade liksom riskhanteringsmodeller och risk processer som vi försöker jacka in i. Då leder inte vi arbetet. Utan då är det dem, men där kan det vara allt från hot analyser till allt. Det är väldigt mycket upp till dem. Hur dem väljer att styra det. Dem kanske har vissa standarder som dem måste följa på grund av den branschen och juridiska skäl och egentligen vad som helst. Och dem har ju gjort det för att det är ett stort bolag och kanske dem har dem här tydligare processerna och strukturerna. Och där har vi också sett att i mindre projekt applicerar du deras riskmodeller på dem så skapar det långsammare processer. Men i dem större projekten hos dem så blir det en mer naturligt del. För då är riskerna en så naturlig del i det hela.
30	D	Långsammare på grund av vad exakt?
31	I1	Långsammare för att man kanske måste följa en process bara för processens skull när den egentligen inte lyfter upp något värde till teamet. Det kanske teamet har redan järnkoll på det här och dem hanterar det, projektledaren hos kunden är fantastiskt, han har järnkoll, han vet riskerna och dem kanske är ganska små.

32	D	När ni jobbar själva så lägger ni mycket tillit på era utvecklare och ert team för att hantera dessa risker. Har det hänt att dem inte har kunnat hantera riskerna på den önskade nivån?
33	I1	Nej, det skulle jag inte säga. Alltså utvecklingsprojekt är alltid utsatt för olika typer av risker. Alltså det är liksom självklart. Vi förlitar oss väldigt mycket på individer och en grupp av människor kan åstadkomma fantastiska grejer. Och då ska de också ta ansvar för det. Alla projekt som sätts ihop, oavsett om de är här eller hos kunden, de har olika faser. I början så kanske de har, de missar. De kanske missar att se en risk som dyker upp två iterationer senare. Men då lär ju sig teamet. Det är en inkörningsperiod för dem också. Och där kan det liksom finnas olika varianter för att du vet ju inte alltid domänen när du kommer in. Du vet inte allting om organisationen. Du vet inte vilka potentiella risker som kan komma. För att det är sånt man lär sig under resans gång. Mycket av dem agila tankarna handlar också om att lära sig mer. Det är därför vi väljer att vara iterativa, för att vi tror att vi hela tiden kan lära oss mer om det vi ska bygga. Och där kommer ju också risk in. I och med vi lär oss mer så får vi en annan syn på risker. Sedan kan du ha liksom tydliga, innan projektet börjar, väl definierade liksom risk modelleringar på ett sätt. Men det är en del av arbetet innan, kanske när första iterationen börjar. När man börjar kanske med att - bryter, göra backlog grooming och så tar man in en risk. Det är en viktig aspekt av ett projekt, precis som du har alltså du kan föra in risk i en liksom estimeringsövning också.
34	D	Anser du att riskhantering har påverkat framgång eller misslyckanden av era projekt, och till vilken utsträckning i så fall?
35	I1	Det är klart det gör, alltså all form av riskhantering är ju till att för att liksom få ett lite mer behagligt resa på ett projekt och att det ska lyckas. Sedan hur man väljer att göra den här riskhanteringen är liksom väldigt projekt specifik eller väldigt kund specifik. Och jag tror inte att riskhantering per sig är något negativt, men graden man gör på den och hur mycket tid och hur mycket process man lägger på det, det varierar så oerhört. Det är precis samma sak som att ha ett utvecklingsteam med två personer. Ska dem köra Scrum? Det verkar idiotiskt i min värld. De sköter all kommunikation mellan varandra dagligen. Och de sitter på samma plats, ska dem ha alla ceremonier? Nä då kanske de behöver något mer kontinuerligt flöde. De kanske till och med kan gå ifrån iterationer. Iterationer är också ett täcken på att teamet kanske inte är fullständigt moget eller organisationen inte är fullständigt mogen. Kan du komma från iterationer och ha den disciplinen så vore det ganska trevligt i många fall.
36	D	Och har ni traditionella eller plan baserade projekt också? Vattenfalls modell eller liknande.

37	I1	Nej, vi jobbar inte med vattenfall eller den sorten av projekt.
38	D	Har du varit med om sådana projekt tidigare?
39	I1	O ja!
40	D	Och hur skulle du säga att riskhantering i de traditionella projekten skiljer sig från de agila?
41	I1	Precis som vattenfall skiljer sig från agila grejer att du inte kan planera allting och försöker du planera allting i oändlighet så skapar du väldigt lite värde. Du lär dig så pass mycket under resans gång. Och precis samma anledning - alltså missförstå mig inte med detta men det finns olika typer av riskmodellering och riskhantering som du kan göra inför ett projekt. Vissa delar av dem kan du göra i ett agilt projekt, vissa delar av dem kan du också göra i ett vattenfallsprojekt. Skillnaden är att risker förändras, vilket man inte tar höjd för igen liksom. Mer gammal traditionell - precis samma sak som att många bolag kanske försöker planera jätte jättemycket innan. Det är klart man ska planera. Och klart man ska se på vad man ska bygga och vilka risker det finns men det finns många agila projekt och större agila projekt, dem har två, tre års planer. Dem har ganska tydliga detaljer. Och det är extremt - alltså kan någon riskmodellera på ett årsplan, då borde dem vara snuskigt rika som troll och sälja böcker och föreläsningar. Men jag tror inte riktigt att man kan det, jag tror att dem hade ljugit för kunden om de hade sagt att dem kan det.
42	D	Tycker du att man behöver mer riskhantering när man jobbar agilt jämfört med traditionellt?
43	I1	Alltså precis som alla andra system, när du jobbar med liksom agilutveckling så allting handlar väl om din mogenhet som organisation i det mjukvara ni ska bygga. Kan du - förstår du inte undermeningen över varför man gör något, om du är en organisation som behöver väldigt mycket stödhjul, då är det kanske att dem följer den processen. Ja då är det kanske bättre att dem har en definierad riskhanteringsprocess också. Att dem har mer formaliserat det för dem behöver dem stödhjulen. För annars så kommer det projektet krascha. Precis samma sak som en organisation som säger - ja men Scrum passar inte oss. Ja men varför då? Nä vi tyckte att morgonmötena var katastrof. Vi är ett team på nio personer, så vi struntar i det. Så man plockar de goda bitarna för man tycker det är det man har tid med. Då har de kanske inte förstått agile. Om du gör ett medvetet beslut, om de säger att - okej, vi är två, tre pers i teamet eller vi är såhär och såhär. Vi behöver inte det här för att det skapar inget värde för oss därför att. Om du tar det medvetna beslutet över din process, då är det en annan sak. Om du däremot inte riktigt kan förklara varför du plockar bort något och du kanske skapar ett problem genom att plocka bort det. Agile kräver extremt mycket disciplin. Och alla organisationer har nog inte den

		<p>mognadsgraden än. Så förhoppningsvis kan de börja med ganska tydliga processer och den typen av projekt har du förmodligen något att tjäna på genom att ha en definierad riskhantering.</p> <p>Men det man kan säga generellt är ju att precis som - ju större bolaget är, ju större kunden är, desto troligare är det att de har någon process eller någon sorts av ramverk för riskhantering som man vill att projekt ska genomgå. Och det kan sättas som ett constraint mot det utvecklingsprojektet vi ska verka i. Då får vi anpassa oss till det. Okej, dem har dem här riskhanteringen, hur kan vi jacka in på det för att följa deras process men ändå vara agila och märker man att - nä, men det skapar inget värde, då får man lyfta upp det till dagordningen och se okej, vad är egentligen anledningen med att ni behöver dem här riskanalyserna? Varför behöver ni den typen av riskhantering? Kan vi jobba på något sätt med det? Eller kan vi göra förändringar i det? Eller varför har ni det? I vissa fall är det - vi har bara den här processen, vi måste ha den för våra certifieringar, eller vi måste ha det för vad det än nu kan vara. Ja då får vi applicera den ramverken organisationen har ändå. Det är liksom inte konstigare än så att applicera deras riskhantering.</p>
44	D	Men i så fall blir det kanske, som du sa tidigare, waste of time. När man måste följa en process istället för att jobba på sitt eget sätt.
45	I1	Ja absolut, det kan det bli. Jobbar du med ett bolag med medicinsk utrustning med människoliv på spel, då är det ganska viktigt att jacka in i deras processer. Även hos kunden så är det kunden som ansvarar för sina processer, och vi ansvarar för att stötta dem i arbetet i deras processer. Men vi förväntas ju också lyfta upp det i dagordningen om vi tycker att något inte skapar något värde. Sedan finns det vissa grejer som måste vara till och det kan vara allt från att det måste - det man bygger måste reviewas av en board för en riskanalys eller något liknande, och det kan man alltid jacka in. Det är extremt kundspecifikt.
46	D	Men jag tror det var alla mina frågor. Finns det något du vill tilläga eller något du inte har kunnat ta upp innan vi avslutar?
47	I1	Jag tror att - är man nyfiken på det så tycker jag att ett bra sätt att börja med det för att få upp det på dagordningen är ett riskregister eller börja jobba med det. Och så formalisera egentligen riskerna i ett projekt initialt. Dem vet hur man hela tiden jobbar mot det. Precis som ett team kanske jobbar med en backlog och teknisk skuld eller vad det nu kan vara. Men att försöka förstå undermeningen med varför agile är riskhanterande. Och sedan försöka kontinuerligt kika på risker. Och omvärdera och utvärdera. Använda root cause analysis om något går åt helvete eller liknande. Det var mina tankar. Sedan kan jag väl säga att vi jobbar ju i förhållandevis ganska små utvecklingsprojekt. Vi har ju vissa projekt där våra konsulter kanske sitter i - alltså vi jobbar förhållandevis ganska små utvecklingsprojekt här ifrån, i vårt kontor. Men sedan har vi konsulter som sitter i projekt med 150

		<p>man. Någon stor SAFe modell där de gör roam, sådan riskhantering osv. Och det finns bra saker med dem också. I många fall märker man att - den sista undermeningen är väl att man ska alltid ifrågasätta varför man gör ett visst steg i en process. Vad skapar det här för värde? För fyller det ingen funktion, ja då kanske man borde ta och fundera på om man verkligen ska ha det, eller om det finns något annat mer lättvikt system om det är en tung process. Det är undermeningen, jag tycker att man bör förhålla sig till.</p>
X	X	<p>After the interview, we continued discussing and a relevant comment was made. It is added here below.</p>
48	I1	<p>Ett moget team och en mogen organisation som utvecklar något, lyfter fram och tar upp risker. Det är därför man är agile liksom. Det är just för att kunna hantera risk. Oavsett om det är pengar eller om - vad det än nu är. Sedan hur teamen väljer att hantera den risken. Vissa projekt som kanske har extremt mycket beroenden eller extrem mycket - det är flera team osv. Då tror jag att många av de största riskerna ändå är organisatoriska. Att i mjukvaruutveckling. Ju större projekten blir, ju fler team det blir desto rent svårare blir det organisatoriskt att hålla liksom spår. Jag har nog aldrig stött på en människa eller en utvecklare eller en projektledare som jobbar i SAFE, Scaling Agile Framework, det är väl agile för storbolag. Och ingen tycker att det fungerar bra som jag har stött på. Alla tycker att det skapar otroligt mycket waste, mycket ceremoni, mycket artefakter. Och man fokuserar inte på vad som ska byggas och vilka problem man ska lösa. Det är väl utmaningen med alla systemutveckling som är stora. Agile är väldigt svårt att skala för att ju mer du skalar det, desto fler organisatoriska risker får du. Den är skitsvår, det finns ingen bra lösning. Och jag tror att, återigen det handlar om vilka människor som är inblandade i det och försöka styra projektet och försöka styra processerna. Med en riskhantering, risk management och till vilken utsträckning det behövs.</p>

Interview 2

Row	Speaker	Answer
1	I2	<p>Vi jobbar mycket med stora business kunder. Lite större företag, förvaltningar, myndigheter och så. Vi har egentligen att man levererar - i vissa fall har vi att man levererar vattenfallsprojekt som vi utvecklar och levererar. Men fler och fler vill köra agilt. Men när man kör agilt så måste man låsa en eller två av variablerna; scope, tid eller budget. En av dem måste vara låsta för att annars kan man aldrig köra agilt. Så i mitt senaste projekt var det tid och budget som var låst och det var scopet som vi justerade för att vi ska hinna med att leverera allt det är vi ska leverera. Men jag skulle säga att kunden inte alltid är agil. Man vill gärna jobba agilt och man vill jobba iterativt, men man är inte beredd att ta dem chanserna eller riskerna som det innebär. För att har du gjort ett projekt och delar upp den i sprintar, då måste man börja plocka bort från en sprint inte följer planeringen, men ofta vill inte kunden plocka bort utan de vill lägga till. Så jag kör alltid risk management, risk planering. Jag vill i början av ett projekt, när man sätter scopet, så även om det är ett agilt projekt så sätter man scopet. Jag startade nyss ett jättestort intranät projekt från en B2B kund, för några veckor sedan. Och då måste man titta på helheten. Vad är det som kan påverka projektet? Om jag ska göra ett intranät, då är ju sponsorn, internal communications och sedan är det kundens projektgrupp sitter i communications och sådär. Men sedan har man alltid då risker med deras IT avdelning som ska hosta miljöerna, som ska leverera den typen av plattformar. Det ska byggas en AD. Och allt detta ska klappa för min tidsplan ska hålla. Så måste de andra projekten och tidsplanen hålla. Och då måste man göra en form av risk kalkylering, en risk management. Alltså att titta på vilka yttre faktorer kan påverka mitt projekt som jag inte egentligen har kontroll över men som jag måste vara medveten om så jag kan handskas med dem riskerna om de skulle dyka upp eller när de kommer. Sedan måste man också, även om man jobbar iterativt så kommer man till det här läget när en sprint inte går som planerat. Eller om man stöter på - vi gjorde ett jättestort extranät för en kund, en e-handelsplats för deras dealer så att säga. Så att de kan gå in och se produkter, de kan läsa ordrar, man kan lägga beställningar, man kan se lagerstatus och sina rekommenderade retailpriser och sådana grejer. Och det var ett så stort projekt så det blev såhär 2500 timmar som ska liksom löpa över väldigt väldigt lång period. Då måste man också göra risker för man vet inte riktig vad man stöter på. Vi stötte på en ny produkt, en commerce katalog som inte fungerade som vår leverantör hade sagt i CMS världen. Och då börjar man dra över tiden och då måste man ha en dialog med kunden. Man måste ha liksom vilka risker finns det, okej nu har vi uppfyllt, nu har vi nått denna risk. Vad har den - för en risk har alltid en impact. Antingen på kostnad tid eller lösning alltså</p>

		budget. Vad är det som för impact, vad innebär det för lösningen? Är det någonting som vi kan säga - okej, då scopar vi ut detta och så lägger vi det i slutet eller i en annan sprint eller annat projekt. Så det är agilt, men där är risk management i det vi gör.
2	D	Så ni har integrerat risk management med era agila processer. Men hur jobbar ni agilt? Är det till exempel Scrum ni följer? Eller har ni utvecklat ert egen agil process?
3	I2	Ja, den är flexibel skulle jag säga. Nä, vi kör Scrum baserad. Så vi kör - det första som händer när vi får ett projekt är att vi tittar på kravspecen, vi börjar estimerar för att få en uppfattning av hur stort det är, och för att vi ska ha förstått kraven. Sedan får kunden godkänna det estimerade eller det priset, antal timmar som vi säger att det kommer ta. Efter det sätter man upp en typ av backlog. Så vi jobbar verkligen enligt Scrum metoden. Så då sätter vi upp en backlog, bestämmer tillsammans med kundens projektledare och team. Det här vill vi ha in i den här sprinten. Vi har börjat köra så att vår första sprint är alltid tre eller fyra veckor beroende på hur stort projektet är. Men om det är ett medium projekt eller ett large projekt, då brukar vi sätta en första sprint, som ska vara ungefär tre fyra veckor för att det ska vara - man vill liksom bygga upp miljöerna, man vill sätta en foundation så det finns något att se och det finns något att testa när första sprinten är klar. Så därför är den ofta - vi tycker att det funkar bättre. Den första sprinten är lite längre, sedan kör vi två veckors sprintar med sprintplanering på måndagar, där man går igenom - vad finns i backloggen? Vad vill man lägga i denna sprint? Hur många timmar har vi? Alltså hur många utvecklare har jag i projektet? Hur många timmar kan jag leverera, vad innebär det rent praktiskt? Och sedan är det daily Scrums varje morgon där man går igenom dessa klassiska - vad gjorde du igår, vad ska du göra idag? Finns det några hinder? Finns det något problem vi måste sätta oss ner och diskutera? För att då kan man - under en daily är det lätt liksom att hitta - vi har fastnat i detta här. Ja men bra, då vet vi att ni har fastnat. Då kan vi boka in ett möte efter och sedan kan vi börja titta på vilka utvecklare eller vem är det jag behöver ha med för att förstå detta här? Är det ett krav som är konstigt? Är det kunden som ska vara med? Så kan man köra separata möten. Men daily ger alltid en temperatur check liksom. Vad gör vi? Vi har valt att bjuda med kunden i daily. Så att vi inte kör internt utan vi har ofta kundens projektledare med. För har man några frågor kan man snabbt ställa och fråga. Och då blir det öppet, och det blir transparent. Ofta kan kunden bara sitta med och vara tyst. Men behöver man feedback tycker vi att det fungerar jättebra. Och sedan då när sprinten är slut så gör man en liten retrospektiv och sedan så diskuterar man retrospektiv. Vad är det vi vill ha med nästa gång liksom. Så det är aldrig så att vi kommer oförberedda till kunden. Jag har alltid en idé. Jag har redan gjort en grov tidsplan. När jag får projektet så sätter jag alltid upp en tidsplan med sprintar och så ungefär vad vill jag ha med. Hur många sprintar kommer

		behövas? Hur mycket kalendertid? För det som är också viktigt är att man har ett estimat, och sedan har man kalendertid. Och de korrelerar inte alltid. Så där måste man också få in liksom som nu, över sommaren till exempel. Då har man inte lika många utvecklare. Du har en period där kunden kanske inte är på plats. Då måste du ha mer kalendertid för att lösa samma saker så att säga. Men jag brukar alltid ha en uppfattning från början till slut, det här vill jag ha in i dem olika sprinterna och så får man se, då har man levererat allt du skulle leverera? Ligger det med något kvar som ska vara med i nästa sprint? Hur påverkar det? Vad kan vi göra så att säga?
4	D	Följer ni ungefär samma process i alla projekt, eller skiljer det sig mycket?
5	I2	Nej, vårt företag har tretton olika kontor. Vi är väl ungefär åttahundra anställda nu. Man gör lite olika på de olika kontoren. Jag saknar från min tidigare arbetsgivare som var mer än renodlad webbyrå. Där hade vi verkligen så att alla jobbade efter samma. Då gick man till och med in så långt och sa att alla projektledare gick en PRINCE2 kurs, man blev certifierade, man började använda den metodiken för att föra - alltså projektmetodiken för att få bättre koll på hur det ska vara i utvecklingen. Och då fanns det templates, det fanns sådana saker. Det kan jag sakna lite sådär. Ibland kan det bli för styrt, och ibland kan det bli för slapt eller vad jag ska säga. Vi är ju här något mellan, så här får man göra lite kontorsbaserad. Vad funkar? Och hur driver vi projekt? Så här skulle jag säga att vi gör på samma sätt, men alla kontor gör inte likadant. Och jag tror att det är så på lite mindre - lite mindre konsultbyrå. Kommer man till de stora så tror jag att det är mer - allting är strömlinje formaterat. Man kör projekt på ett sätt liksom. Men nej, inte här.
6	D	Och vilka risker är mest förekommande under era projekt?
7	I2	Svår fråga. Det beror på projekten, det beror på - jag körde ett commerce projekt, det här intranätet. Det var ett commerce projekt. Det var det första commerce projektet vi körde här. Och då hade vi - dels hade jag en stor risk budget ifall estimatet inte skulle hålla. Så jag hade en buffert budget för jag fick lov att - liksom om jag hade estimerat på 1500 timmar så hade jag kanske 500 timmar till som en liten buffert för att det var nytt, oprövat mark sådär. Men det jag skulle säga är att det beror lite på komplexiteten och kunden. Vilka risker man har, om man - alltså det är det här yttre påverkan som nu det nya projektet, hos den nya kunden som jag är på väg in i, att där är - när man levererar till stora företag som har såhär klassisk organisation med en egen IT avdelning och som har mycket sådana processer kring supply chain och avtal, och det ska skrivas hit och dit för att man ska få suppliers och så. I sådana organisationer är där alltid beroende av andra projekt. Hur många projekt kör kunden? Är de medvetna om att detta kommer kräva resurser från deras sida också?

		<p>De är sådana klassiska risker som finns. Och sedan skulle jag säga att estimering i konsultbranschen är svårt. Jag har ju en faktor egentligen, när man har estimerat, hur mycket måste man gånga det med för att estimatet ska hålla? Det lär man sig över tid. Det är väl sådana klassiska liksom. Och sedan scope creep, det är mer - jag gör någonting, sedan diskuterar jag scopet med kunden. Kravspecen liksom. Vad innehåller kravspecen? Vad står där faktiskt i dokumentet? Vad är det vi har sagt att vi ska leverera? Det här förarbetet är jätteviktigt att när man tittar på kravspecen så måste man läsa igenom den och förstå den. Man måste vara tydlig med - vi har tolkat detta såhär och såhär. Det är det vi kommer leverera. För kunden kommer alltid komma på att - ah, orderhistorik, det har vi inte med där men det måste finnas liksom. Och läser man här, order fullfillment - ja men då ingår det i den. Men det gör det inte, alltså du vet. Den här diskussionen med vad är ett scope creep och vad är en bugg. När man har levererat en lösning och de ska testa det, så är där alltid lite problem. Då är det frågan, var kommer problemet ifrån och speciellt i sånhäre komplexa miljöer där man gör projekt. Sedan så deployar man in till en produktionsmiljö. Och så är det något i produktionsmiljön som smäller. Och vi kör ju - när vi utvecklar så har vi våra utvecklingsmiljöer- de är egentligen utvecklingsmiljöerna som varje konsult har liksom. Sedan deployar de upp till en test miljö som är hyfsat lik kundens produktionsmiljö, men man testar lite grejer där. Det är inte klockrent. Man får lov att - fortfarande får man lov att leka lite runt i testmiljön. När vi deployar från test till vad vi kallar stage, så är det en sådan pre-production miljö som ska vara exakt lik produktionsmiljön. Där ska vi testa allt det bara går att testa. Och är det väl testat där så ska det inte smälla när man väl går ut i produktion. Men det händer ju ibland. För det beror ju på kundens test tid. Hur mycket tid har de lagt ner, hur mycket vill de betala för test? Och så i ett projekt där man gör ett estimat så - man skulle vilja ha en testledare, och testcase och en testare. Men det är den första kunden egentligen säger nej tack till. Och det är för att det tycker att de kan testa själva. Det finns ju use cases på vad det är man ska leverera. Som en användare vill jag... Som en användare ska jag... Det finns ju use cases som supporterar tasken som vi utvecklar alltså det som vi har brutit ner. Man kan använda use casen som test case, men det blir inte riktigt samma sak. Ett test case ska egentligen täcka in så mycket mer. Och där fallerar det ofta. Så test perioden och buggarna, eller scope creep är det som oftast missas skulle jag säga.</p>
8	D	Och har ni någon specifik åtgärd till detta? Har ni någon specifik process som ni använder för att hantera dessa risker?
9	I2	Scope creep, där får man bara ha mycket skinn på näsan. Det gäller att kunna bråka, att på ett fint sätt kunna förklara för kunden att det inte går. Och sedan test är ju svår. Vi brottas fortfarande med den. Vi tittar mycket på unit test, automated tests för att underlätta för oss. Men det ersätter aldrig en full testning, en regression testning av en lösning så att säga. Vi

		har infört - i vår organisation har vi infört en test management avdelning där man ska börja testa. Men det är framförallt några kunder som sa till exempel eller till myndigheter och sådana, där det är mer de kraven, de vanliga dagliga kunderna har kanske inte de kraven på test på samma sätt. Utan de testar gärna själva.
10	D	Kan du beskriva lite hur er risk management process ser ut? Hur ni brukar hantera risker.
11	I2	Ja! Ofta börjar vi med en workshop där man egentligen bara listar alla risker. Och då har vi gjort en brainstorming innan eller förslag på topics liksom. Finns det om man säger, ska vi leverera ett intranät, ja men då kan ju intranätets innehåll vara till exempel en risk, att det kommer inte skapas innehåll i tid. För det sitter en kommunikationsdirektör på toppen som äger projektet så att säga. Och sedan listar man riskerna med miljöerna, med utvecklingen osv. Sedan börjar man prioritera så vi har allting i en - man prioriterar alltid riskerna från ett till fem liksom. Vilken risk har störst sannolikhet och infall? Och får man då en etta till exempel, då måste man sätta dem på åtgärder med det samma. Sedan tittar vi på impact, om den här risken skulle inträffa, vilken impact har det på projektet eller på det vi ska leverera. Och sedan tittar vi på severity, alltså allvarligheten i risken i förhållande till att om det inträffar, hur allvarligt är det? Och sedan gör man en beräkning på de tre för att se risk status så att säga. Och sedan sätter man en ägare och en kommentar alltså från början. Vad gör vi om den här risken skulle inträffa? Man redan har tänkt att om den här risken inträffar, då måste vi detta och detta. Och när den inte då inträffar och faller ut, då vet man exakt, det är du som äger risken, det är du som ska börja jobba.
12	D	Vem brukar vara ägaren för risken?
13	I2	Det är projektmedlemmarna. Jag äger - som projektledare äger man oftast riskplanen tillsammans. Vi jobbar så att vi alltid har i dem projekt vi levererar - har vi alltid en projektledare internt. Sedan brukar kunden ha en projektledare. Och det egentligen är vi två tillsammans som äger riskplanen. Men det kan vara vem som helst i projekt teamet som äger risken. Som nu hos kunden så är ju sponsorn är marketing and communications vice president. Hon som äger projektet är kommunikationsdirektör. Sedan är det ett projektgupp där som vi har lite olika medlemmar. Och då kan ju vem som helst av dem äga risken.
14	D	Och den här processen, skiljer det sig mycket från projekt till projekt?
15	I2	Nej, det är i princip ganska likt. I ett litet projekt kanske man inte behöver titta så mycket på impact och severity. Utan då ska man mer lista riskerna. Jag hade ett projekt där precis - vi skulle gå live med en ny applikation, alltså en som skulle ha påverkan på plattformen. Och så hade

		<p>vi ett problem, och så gjorde vi en ny riskplan alltså, lista fem risker. Om de här fem riskerna, vad händer om de faller in. Ett till fem. Och så en åtgärd. Här kommer vi en dag och så måste vi plocka ut det. Vi får berätta för kunden att vi inte kan leverera, vi måste plocka ut den liksom. Men då hade vi bara behövt att göra den mindre, men jag har alltid den i grunden och sedan beroende på vad det är, på ett litet projekt kanske man inte behöver gå hela vägen. Så det beror sig lite på organisationen också. För att jag har mina eller vi har våra projektmetodiker. Vi har vår riskhantering. Men kommer vi till en kund som - jag har jobbat lite med större etablerade företag. När man kommer till en sådan organisation så har de ofta sin projektmodell och sin riskhantering. Och då gäller det att, då får man få använda kundens modell. Men har man använt en innan eller om man vet ungefär hur man arbetar så brukar det inte vara så stor skillnad att byta modell. Och ibland kan kunden kräva att - vi vill att ni arbetar på detta sätt, och då får man bara anpassa sig.</p>
16	D	<p>Finns det några artefakter eller tekniker som ni använder för att hantera riskerna? Till exempel en risk burndown chart eller liknande för att hantera riskerna lite lättare?</p>
17	I2	<p>Nej, inte en risk burndown chart. Vi har ju för backloggen och alla våra tickets så använder vi ett system som heter Jira, och där kan man se liksom burndown rate på timmar och tickets, hur mycket vi lagt och sådana grejer. Men det är mer för att följa progressen i projektet för att se om vi levererade det vi ska och om vi håller vårt estimat. Det är inte så mycket i förhållande till riskerna, det är mer till liksom den här tiden, scope, budget. Vad är det som är låst och vad är det jag måste förhålla mig till? Och så kan man räkna på vad det är. Ofta säger kunden att tiden, det ska vara levererad till augusti. Budgeten den är ofta låst. Scopet är ofta låst. Ja, okej tiden var flexibel kanske. Vi säger augusti men skulle det bli december så är det inte hela världen. Vi hade nu till exempel en hård deadline i maj. Den var non-negotiable. Den skulle gå live den 31:a maj, det finns andra grejer som det påverkar att det ligger live. Budgeten är fast, så det är scopet vi ska ändra. Och det är det svåraste, för kunden vill aldrig ändra scopet! Det är väldigt svårt.</p>
18	D	<p>Hur upplever du att er nuvarande riskhanteringsprocess är? Är det hjälpanande eller bristande? Är det redundant i och med ni jobbar agilt?</p>
19	I2	<p>Jag tycker att vi skulle dra nytta av som bolag att standardisera våra processer lite. Jag gillar när man har ordning och reda. Det ligger kanske lite på projektledaren, men jag skulle vilja att vi verkligen hade en folder som heter risk och så ligger alla risk dokumenten där. Dem som heter förstudie, dem som heter projekt. Alltså så man hade mallar och templates på allting. Alla projektledare har det, men vi har liksom inte</p>

		harmoniserat det till 100 %. Det hade jag tyck varit bra. Så vi har väl inte en riktig process egentligen.
20	D	Tycker du att man behöver en sådan process när man jobbar agilt?
21	I2	Jag skulle säga att man aldrig klarar sig utan. Det måste finnas lite risker ändå. Även om man kör agilt, om vi är agila men kunden inte är agil, och vi är inte 100 procent agila heller. Vi har fortfarande många utvecklare som kämpar lite med det här med vad som ska in i en sprint och hur man flaggar för man går över tider och lite sånt. Så jag skulle säga att det finns väl ingen som jobbar 100 procent agilt. Man jobbar flexibelt agilt i min erfarenhet.
22	D	Hur skulle du säga att era riskhanteringsprocesser påverkar era agila processer? Blir ni mindre agila för att ni följer några processer?
23	I2	Jag tror det hjälper. När man tar sig tid att göra risk hanteringen från början så hjälper det den agila utvecklingen också. Jag skulle säga att det beror på storleken av projektet. Gör du bara en extern webb så har den inte så många påverkan, alltså om man tittar på en kanal strategi, hur många system har kunden och vad är det kunden gör. Vad använder man de olika kanalerna till och var har de påverkan. Ett intranät måste ha större riskhantering än en externwebb. I ett e-handelsprojekt måste man väva in jättemycket riskhantering för där är så många integrationer. Ett e-handelsprojekt så ska du integrera till kundens system, och produkt informationssystem. Du ska kanske integrera till kundens miljö, alltså standard system, ekonomi system. Kanske ska integrera till ett annat system, och det kanske ska gå via BizTalk APIer och så. Så lite beroende på hur komplext lösningen är. Så även om man jobbar agilt så - jag är kanske gammal modig, det kan vara så, men jag tycker det är bra med en process.
24	D	Finns det några projekt där ni inte använder riskhantering, där ni kanske inte behöver använda riskhantering?
25	I2	Nej, kanske i väldigt små projekt. Gör du ett litet projekt i 300-400 timmar så nej. Det är en så snabb leverans, det tar bara två månader eller något. Så i ett litet projekt, så nej, det skulle jag inte säga.
26	D	Och något som vi inte hann ta upp tidigare, hur ser era projekt ut? Är de stora och komplexa, eller är de varierande?
27	I2	De är varierande. Jag har hamnat i de stora komplexa för tillfället, men de är varierande. Vi har små, snabba, medium projekt, stora projekt. Eftersom vi jobbar mycket med B2B så är det stora projekt.

28	D	För att få en känsla av hur stora projekten är, hur många brukar jobba på dessa projekt?
29	I2	Ja, ett litet projekt kan vara tre pers liksom. Ett stort projekt, jag säger att man inte kan ha ett projekt team som inte är större än - internt hos oss, åtta personer. Har du ett projekt team som är större så har du svårt för att kontrollera det. Då har du svårt för att - jag säger att ett ideal projekt team är sådär fyra, fem personer. Även om det är ett stort projekt så har du lätt för kontrollen och dialogen. För ibland när vi kör stora projekt, så börjar jag som en projektledare. Sedan har jag en Scrum master daily, för att han eller hon ska ha en chans och handskas med det, så kan inte projekt teamet vara större än max åtta pers. Har man ett större projekt så måste man nästan dela det i två ben.
30	D	Och du nämnde att om projekten är små så behöver man kanske inte riskhantering. Finns det andra faktorer eller omständigheter där riskhantering inte skulle behövas?
31	I2	Om kunden är väl känd. Att man har jobbat länge med en kund. Alltså om man har ett långt partnerskap med en kund och man har jobbat länge med dem, och kan lösningen och vet vad det är som ska levereras, då behöver man kanske inte ha riskhantering. Men då måste det vara liksom boxad nästan, att du levererar detta, och så gör man det. Men då måste kunden vara välkänd, och miljön måste vara välkänd. Då måste man nästan ha varit med från början och utvecklat plattformen liksom.
32	D	Anser du att riskhantering har hjälpt er, eller har det haft sina nackdelar?
33	I2	Jag kan inte se någon nackdel med riskhantering. Jag kan tycka att även om man inte följer den, alltså ibland kan det hända att riskhanteringen blir en pappersprodukt tills dess att du skriver in ett problem. Men fördelen med riskhantering är att det blir lite fritt brainstorming kring vad skulle kunna gå åt skogen. Vad är det som skulle kunna bli fel. Och då kan man på det sättet också, när man diskuterar och har workshopen tidigt, det kan vara riskhantering, det kan ta tolv timmar eller fyra timmar eller en del av en workshop, ett uppstartsmöte hos kunden. Vi kör lite - vad har vi för risker med det här projektet, och så listar man riskerna. Bara det gör ju att, när man då ska lista risker så kan man få förståelse för - har vi förstått projektet eller har kunden förstått vad vi har sagt när vi säger att vi ska leverera något. Så det ger en annan dimension till - har vi förstått vad det är vi ska göra, har kunden förstått vad vi ska leverera. Ibland kan man tycka att detta är kristallklart, ja men vi har det här kravspecen. Sedan har vi gått igenom det, vi har estimerat det och så talar vi med kunden. Sedan börjar vi jobba och inser att - nä, kunden och vi pratar inte samma språk. Jag har uppfattat det på det här sättet och har skrivit det här. Kunden säger att nej, det är såhär och såhär och har uppfattat det annorlunda. Och då måste vi ändra om. Det har med

		erfarenhet att göra. Det kan hända, men när det händer så kan en risksektion vara det som liksom där man börjar lista liksom, vad har orsakat det här? Vad är det för problem? Då kan sådana grejer komma fram. Att man inte tänker på samma sätt, eller man har missförstått varandra eller så.
34	D	Har ni några traditionella, plan baserade projekt, likt vattenfallsmodellen?
35	I2	Ja, företaget har sådana projekt. Ett av våra kontor kör sådana klassiska, vattenfalls projekt.
36	D	Har du också varit del av sådana projekt?
37	I2	Nej, jag har inte varit det den senaste tiden. Den senaste tiden har jag nog - fast jag säger inte agilt, jag säger Scrum baserad, flexibelt. Alltså det går så mycket mode ord i agilt, men det blir aldrig riktigt agilt men det blir - jag gillar att arbeta i sprintar med tydliga leveranser för då kan man - om du jobbar i ett vattenfall, och tar hem ett projekt på 700 timmar, alltså har du då dialog med kunden under projektets gång om vad ni gör, då kan de få se något. Men du vet aldrig riktigt om - jag tycker riskerna blir för stora med ett vattenfallsprojekt än vad det är när man jobbar agilt. Jobbar du med sprint så kan du ju snabbt se om det ligger på spåret eller inte. Kunden kanske förstår lösningen, kanske kan göra lite mer buy-in på lösningen än i ett vattenfall.
38	D	Blir det lättare eller svårare att hantera risker när man jobbar agilt jämfört med traditionella?
39	I2	Jag skulle säga att det blir lättare.
40	D	Och skulle man behöva mer eller mindre risk hantering när man jobbar agilt, jämfört med traditionella projekt?
41	I2	Mindre risk hantering när man jobbar agilt, men du kan inte utesluta den tycker jag. Men jag tycker det är högst personligt. Vi hade en som kom och gav en föreläsning på Scrum och agila metoder, och varför de inte fungerar. Vad är det som gör att alltså så. Fantastisk människa. Ibland du vet jobbar man agilt så har du din lilla bild med backloggen och refinement och sånt. Och du fastnar i olika saker. Och beroende på teamet. Har du nu fått som jag har en rätt så inkört team. Jag har ett team på tolv pers. Kan du hålla ett fast team på tolv pers och så använder du dem till olika projekt, då lär du känna dem bättre och hur de fungerar, och hur du kombinerar ihop dem. Men när du har ett team på nytt, eller någon som är ny. Då skulle man hitta balansen i teamet igen, för det handlar mycket om gruppdynamik i teamet. Det är lättare att jobba agilt när teamet känner varandra och har jobbat ihop länge än ett nytt team. För då har du hela den här också balansen mellan liksom vem som kan vad, och

		hur mycket måste jag visa? Vet vi vad vi gör, och utvecklar vi på samma sätt alltså så. Där kan bli lite så social - jag skulle säga att det blir lättare att jobba agilt med ett team som är väldigt nära.
42	D	Behöver man ändå ha standard processer när man har ett mogen och erfaren team?
43	I2	Ja, jag tror det. Det är alltid något som är nytt, så. Jag hade gillat att ha fasta eller tydligt planerade processer. Jag tror att man tjänar - det är alltid så att när det ska in en ny projektledare eller du jobbar med en ny kund och så ska du förklara - det är alltid något som är nytt. Och då tror jag att det hade varit bra - jag hade tyckt det hade varit skönt - jag har bara varit här i lite mer än ett år. Jag hade tyckt det var skönt när jag kom hit om det verkligen hade funnits en - såhär jobbar vi, en mall. Men nu kom jag hit och fick lite så - och sedan får jag sätta nästan sätta scopet lite också. Men jag tror jag får börja inse att jag gillar processer! Jag tror det underlättar, även för utvecklarna att veta att någon - för man följer en viss process, en standard process. Sedan kan man göra den - du vet mer komplex, större eller mindre. Den kan vara flexibel. Men så länge du har en standard process att följa och alla vet att det är såhär. För att leverera detta så gör vi såhär. Och sedan beroende på hur stort eller litet eller komplex projektet är så kan man kanske ta ner vissa faser. Men jag tror på en process liksom, en botten. Så blir det flexibelt. Nyckelordet här är flexibelt.
44	D	Jag hann inte ställa frågan under intervjun, men vad är din roll i företaget?
45	I2	Ja just det. Jag är ju digitalstrateg och projektledare. Jag var ju med en konceptutveckling, digital strategi för företag liksom. Projektledning är ju mer - nu gör vi olika projekt hos kunden. Men vi vill gärna ta på oss ett helhetsåtagande så när man tittar på kunden, vad gör de rent digitalt? Vad ingår i en digitalstrategi? För det är liksom så flummig begrepp höll jag på att säga, men det är ett rätt så brett. Så tittar man på vad är det ni vill göra, vilka initiativ finns det, och vad är det som driver den digitala mognaden? Kan man koppla på det till business value? Om man gör ett webbprojekt eller så, då har det påverkan på affärsprocesserna och då ska säljarna arbeta lite annorlunda. Customer service ska arbeta lite annorlunda. Och då tycker jag - jag gillar det här med agilt att man tittar på helheten, försöker styra upp det och försöker visa att - det här är en del av affärsstrategi så och där på strategi nivån har vi ett par projekt, och det är ofta så när du jobbar långsiktigt med kunder. Men annars är jag ju projektledare också. Jag hade hellre varit bara strateg, men det kan man inte vara. Man måste ha lite andra projekt också.
46	D	Hur länge har du varit projektledare?

47	I2	Jag har jobbat i 17 år. Så jag gick ur universitetet 99-2000 och kommer från kundsidan. Så jag började som affärsutvecklare, sådär business developer. Sedan började jag tycka att krav och kund relationen var roligare, alltså projektledning. Så gick jag på projektledarrollet. Jag tycker jag blev en bättre projektledare än utvecklare. Framförallt nu. Nu har jag varit projektledare i tolv tretton år.
48	D	Då tror jag att jag har ställt alla mina frågor.
49	I2	Ja, och det kan vara skillnad om du - nu har du ju pratat med mig. Vi jobbar ju mycket med webb baserade projekt alltså e-handelslösningar, intranät, extranet, externa webbar. Skulle du träffa någon som kanske jobbar mer renodlad industri med företag - du vet databassystem och andra typer, då kan det hända att de har lite mer fyrkantiga skulle jag säga - alltså webb världen ändå är lite mer flexibel, men det beror sig på vem man intervjuar kanske. Vad man har för inriktning på det man gör. Jag tror jag har fått min rikshantering med mig från - dels från när jag jobbade med på kundsidan, för då jobbade jag med ett stort internationellt företag. Och sedan nu att jag jobbar mycket med stora B2B kunder. Då kommer det här som ett brev på posten, att det finns alltid risk hantering och sådär i dem, hos kunden så att säga. Projekt metodiken, projekt metoden och sånt.
50	D	Finns det något du vill lägga till eller ta upp innan vi avslutar?
51	I2	Nej, jag tror inte det! Men om du vill komplettera med något så får du bara höra av dig liksom.
52	D	Ja! Då tackar jag så mycket för mig.

Interview 3

Row	Speaker	Answer
1	D	Som en introduktion, vad är din roll i företaget?
2	I3	Min roll i företaget, jag är lite olika saker men en form av agil projektledare ibland, ibland lite mer traditionell projektledare och ibland renodlad Scrum master, agil coach. Det är främst det. Sedan har jag också utvecklarbakgrund också och sitter ibland och gör utveckling här också.
3	D	Hur länge har du jobbat med agila projekt, som en agil projektledare?
4	I3	Sedan 2010 ungefär.
5	D	Du nämnde Scrum, jobbar ni enligt Scrum, eller har era agila projekt olika former?
6	I3	Det är olika former. Vi har väl nästan alltid lite custom per projekt. Jag skulle säga det är rätt sällan det är ren Scrum. Det är nog oftast att det är någon variant av antingen Kanban eller Scrum där vi inte alltid kör sprintar med en planering innan. Men vi kör oftast sprintar i form av att vi har en demo och en leverans. Det är inte alltid vi planerar upp liksom det här är det sprint backloggen, utan det kan ibland vara flexibelt beroende på projekt.
7	D	Varför väljer ni att göra ändringar till Scrum?
8	I3	Det är till exempel för att vi inte har en committad sprint backlog, det vill säga det här ska vi göra sprinten utan vi vill kunna vara mer flexibla och ändra saker. Annars att ändra saker inne i sprint backloggen i Scrum är lite mer sådär undantagsfall eller undantagstillstånd. Det har hänt någonting. Vi ser det mer som det händer hela tiden. Så därför vill vi kunna vara flexibla och kunna ändra scopet under sprinten.
9	D	Ser varje projekt annorlunda ut? Finns det någon core agil process som ni följer för de flesta projekt?
10	I3	Nej, coren är att vi kör agila projekt. Det enda jag menar med det är att vi följer agila principer. Vi gillar det agila manifestet och har ett agilt mindset. Sedan i varje projekt är det upp till dem att välja hur de ska implementera dem här tankarna och vilka verktyg man ska använda.
11	D	Jag förstår. Och som en bakgrundsfråga, vilka typer av projekt brukar ni jobba med?

12	I3	Det är allt möjligt, det är högt och lågt. Det är alltifrån små mobilprojekt där vi bygger en app där det är bara några stycken till lite större projekt där vi bygger massa servrar med koppling till olika system och integrationer och betalningslösningar osv.
13	D	Och för att få en bättre förståelse, hur många brukar jobba i dessa små och stora projekt?
14	I3	I de små ungefär 2-3 personer. Och som mest blir det kanske 10. Och då är det mycket.
15	D	Och jobbar ni bara här, eller jobbar ni ihop med andra kontor?
16	I3	Vi sitter distribuerat i olika kontor. Vi samarbetar med folk som sitter i olika projekt team.
17	D	Orsakar detta problem? När ni jobbar distribuerat?
18	I3	Ja, det skapar alltid problem att sitta distribuerat, men det är många fördelar också. Så det är värt att göra det.
19	D	Vilka sorts problem kan uppstå i sådana fall?
20	I3	Det kan vara kommunikationsproblem, att det inte alltid är lika enkelt att dela information. Man hör inte saker hela tiden under arbetet dagligen. Det kan vara problem, med att få till en bra whiteboard session om man sitter på olika platser. Att kunna gå igenom arkitektur och hur man jobbar, hur man ska bygga saker.
21	D	Alltså drabbas den informella kommunikationen när ni jobbar distribuerat.
22	I3	Precis, men that said, finns det många fördelar också. Så att det är något vi kommer vilja göra mer och folk vill kunna jobba var som helst. Och vi vill kunna utnyttja deras kapacitet. Det finns folk lediga vid olika kontor.
23	D	Om vi går vidare till risk management, hur brukar ni jobba med riskhantering i era agila projekt?
24	I3	Det är olika. Vi har nog nästan alltid något inslag av risk management. Jag skulle vilja påstå att det är en del av agila, alltså det innefattas av att jobba agilt, att ta hänsyn till risker. Men att man också kan bli inspirerad av med hur man jobbar med risker i mer traditionella projekt. Och vi gör nog både och.
25	D	Kan du ge några exempel från de tidigare projekt du har jobbat med där ni har använt er av risk management i agila projekt?

26	I3	<p>Ja, en sak jag tycker är väldigt viktigt när man jobbar agilt och med risk just det är att man tar hänsyn till risk i backloggen och har en risk adjusted backlog. Att man ser inte bara business value i form av kanske värdet för slutanvändarna, utan värdet utifrån att kunna minska en risk. Så att man kan lägga in refactoring till exempel. En sådan klassisk grej som kan ibland vara svårt att prioritera. Men det kan man lägga i backloggen. Genom att göra denna refactoring så minskar vi risk. Det har ett värde för att sänka risk. Och därför kan vi värdera och jämföra det med andra saker i backloggen också.</p> <p>Sedan har vi ofta risk workshops. Det är väl nästan alltid initialt. Och det är väldigt traditionellt sätt, gör det tillsammans med kunder. Precis som man brukar göra på traditionellt, man tänker på olika risker, rankar dem och bedömer sannolikhet och impacten för oss. Sedan i vissa fall försöker vi följa upp riskerna löpande, där vi går tillbaka till dem och tittar om de här har minskat, om vi har jobbat aktivt med dem. Det kanske vi inte är lika duktiga på att göra med alla projekt. Det är något vi strävar efter att göra mer i alla fall. Vi har också i vissa fall haft risk burndowns, där vi brukar visualisera hur riskerna minskar.</p>
27	D	När behöver ni inte använda risk burndown?
27	I3	I många fall, så är det att där att det finns risker som är kopplade som vi inte kan göra så mycket åt. Det kan vara beroende till andra framför allt. Där det liksom inte är i teamets makt att göra något åt det. Och då är det ingen större idé att visualisera dem om man inte kan göra något åt dem ändå.
28	D	När ni har identifierat riskerna med till exempel er backlog eller ett riskregister, där ni dokumenterar till exempel, efter att man har utvärderat en risks impact och likelihood och fått en exposure rate, har ni några färdiga lösningar eller färdiga åtgärder? Som till exempel man ska mitigera denna risk, man ska delegera den, man ska undvika den osv?
29	I3	Precis. Så blir det ju, och det blir ofta en aktivitet efter att man har haft en risk workshop liksom, okej det här måste vi faktiskt göra något åt. Och då blir det ofta ett backlog item av det. Vad det nu skulle kunna vara för någonting. Jag vet inte, sätter upp redundanta servrar till exempel.
30	D	Och finns det vissa personer som är ansvariga för riskerna ni identifierar, eller hur sköter ni det?
31	I3	Vi har det inom teamet, så det är sällan folk som är ansvariga. Vi försöker köra väldigt mycket cross-functional teams. Så alla ska kunna jobba med allting inom ett team. Och då lämpar det sig inte så väldigt väl att det

		finns en sådär risk ansvarig. Men ifall det är någon som håller i processen kring risker så är det väl Scrum master eller projektledare.
32	D	Brukas det hända att du blir ansvarig för riskerna som Scrum master eller projekt ledare?
33	I3	Ja, jag ser till att det görs en riskworkshop och att man tar hand om det efteråt, att man går tillbaka och tittar på det då och då också.
34	D	För ett allmänt känt problem med agila projekt är att ingen tar ansvar för någonting.
35	I3	Ja, allas ansvar, ingens ansvar. Så kan det ibland vara, men det gäller att man har bra medarbetare som faktiskt känner ansvar. Och det funkar bra för det mesta. Alla har sin spetskompetens. Det kan vara en senior backendutvecklare till exempel som säger -okej, jag ser en risk här. Detta är något vi borde göra något åt, så jag lyfter detta här. Och sedan får man förstås prioritera det tillsammans med kund och produktägare. Är detta något som vi andra tycker som är sannolikt. Det måste komma litegrann från dem som har kunskap om riskerna. Men får man göra något med det så blir det den personen som förmodligen ansvarig för det, eller ansvarig om man ska säga så. Även om vi sällan uttrycker det så formellt.
36	D	Du nämnde det här med backloggen, och risk burndown chart osv. Finns det andra specifika artefakter som ni använder när ni jobbar agilt för att hantera eller övervaka risker?
37	I3	Jag tror inte det. Vi försöker se det som allt annat arbete.
38	D	Och hur upplever du att er nuvarande riskhantering är? Är det bristfällig? Är det redundant när ni jobbar agilt? Är det hjälpsam?
39	I3	Det kan alltid bli bättre. Så vi bör förmodligen jobba lite mer med riskerna. Men jag tycker det är bra att ha det som inställning att detta är något som passar med agile och det är något man prioriterar också. Det är inte alltid man vill arbeta med risk om man inte bedömer att det ger ett väldigt stort värde att arbeta med dem. Så det är prioritering hela tiden.
40	D	Anser du att man måste ha risk management när man jobbar agilt?
41	I3	Jag tycker att man bör ha riskhantering även i agila projekt, men att man kan använda dem agila verktygen och mindsättet. Att lägga det i backloggen till exempel så att man prioriterar mot andra saker som ger värde. Inte se det som en separat aktivitet "nu ska vi ha riskhantering i denna sprint och bara jobba med risker" utan införa det på ett naturligt sätt. Okej, här har vi en grej som vi behöver göra. Det är en risk. Okej, då gör vi det. Då prioriterar vi det så det kommer med i nästa sprint.

42	D	Så ni har integrerat risk management med agila projekt och agila metoder.
43	I3	Ja precis, så att man integrerar dem och gör det inom det agila.
44	D	När ni gör den här integrationen, händer det att ni förlorar värde från de agila metoderna? Att ni brister mot principerna att jobba agilt?
45	I3	Jag tror inte det, nej. Det är inget i Scrum till exempel som säger att du ska ha en risk workshop. Sen kan man ha risk workshop ändå, men jag bryter inte mot Scrum eller någon agil princip så. Men man har med dem här agila värdena. Det är människor som samarbetar och inte bara en massa processer och verktyg som ska styra. Och onödig dokumentation försöker vi också undvika. Det är sällan vi har en separat risk register, utan i så fall är det inne i backloggen. Om man har det så tror jag att man börjar tunga på de agila principerna.
46	D	Om man har för många processer?
47	I3	Om man har för många separata sidospår som står utanför det vanliga arbetet. Då blir det plötsligt inte liksom customer collaboration over processes and tools liksom.
48	D	Finns det några omständigheter där ni inte skulle behöva använda risk management i era agila projekt? Vilka faktorer skulle spela roll i detta?
49	I3	Jag tror man behöver tänka risk hela tiden i allt arbete. Även om man bara sitter och ska bygga ett GUI till en mobilapp, så bör man tänka risk hela tiden. Vad kan hända här om det går fel, och vad kan det bli för effekt för slutanvändarna. Så där är risk i allting.
50	D	Oavsett hur stor projektet är eller hur många som jobbar med projektet?
51	I3	Ja, det tycker jag. Risk bör genomsyra liksom arbetet hela tiden. Mycket av det kvalitetsarbete vi gör handlar om risk också. Vi har något som vi kallar för QA workshops, som är något en av våra medarbetare har tagit fram bland annat. Ett sätt att analysera, det är lite som ett sorts sprintplanning, fast liksom när som helst för en viss story man ska börja på. Passar bra mot Kanban liksom. Och då är det just att tänka på risk, att tänka på vad händer här, vad finns det för fel, vad kan gå fel, hur hanterar vi fel? Så det är typ en risk management också.
52	D	Du nämnde att ni har projekt med 2-3 personer, i mindre projekt. Har ni även då till exempel en risk backlog eller artefakter för att hantera riskerna.
53	I3	Nej, vi har ingen risk backlog. Det har vi inte i de större projekten heller. Utan snarare det är inne i den vanliga backloggen. Ingen separat backlog.

		Det kan hända, men i så fall är det oftast att man vill belysa resultaten av en risk workshop till exempel. Men då är det mest för kundens skull. Här är 10 grejer, detta har vi kommit fram till det är olika typer av risker. Detta är resultatet av vår risk workshop. Men den hamnar i backloggen eller inte alls liksom. Är det saker som är osannolika, oviktiga så kastar vi dem i så fall, om de inte är ens värt att stoppa in i backloggen.
54	D	Till vilken utsträckning anser du att riskhantering hjälper agila projekt att nå framgång? Hur skulle agila projekt påverkas utan riskhantering?
55	I3	Låter man bli att tänka på risk så blir det ju att man bygger saker som fungerar dåligt under olika omständigheter. I vissa fall vill man till och med jobba så, då man tar fram en proof of concept till exempel. Då kan man strunta i väldigt mycket. Om det är något som kastas. Men det är ett val man gör då. Vi ignorerar väldigt många risker. Det finns andra typer av risker som man tar hänsyn till då istället. Till exempel risken att det här inte blir en fortsättning av det projektet liksom. En annan typ av risk också. Och då kanske det handlar om att man gör en proof of concept, man ska visa på något visst resultat som man vill sälja in något, att detta är en bra idé, det som vi gör just nu. Då får man fokusera på de vinklarna på det hållet.
56	D	Och vilken riskattityd brukar ni ha i era projekt? Är det att man ska försöka undvika risker så mycket som möjligt, eller kan ni tolerera risker, eller hur ser det ut?
57	I3	Vi kan tolerera risker. Det måste prioriteras hela tiden. I vissa fall kan man säga, här är det risk att det går sönder. Men, just nu kommer vi ha begränsat antal användare och vi vet inte ifall detta är en bra idé över huvud taget. Vi håller på att utvärdera det. Vi bygger en meanwhile produkt och det är inte alltid som man vill ta hänsyn till allting så tidigt.
58	D	Som ett sidospår, du har jobbat med traditionella projekt också. Anser du att man behöver mer riskhantering i traditionella projekt, eller i agila projekt?
59	I3	I traditionella projekt försöker man planera väldigt mycket upfront, att tänka ut alla problem innan de har inträffat. Det fungerar sällan bra därför att det händer saker hela tiden som man inte hade kunnat förutse. Men om man försöker sig på det, så gäller det att komma på alla risker man bara kan i förväg så att man kan lägga en plan för hur man kan undvika dem. Därför blir det oerhört mycket viktigare med upfront riskhantering och risk identifiering i ett traditionellt projekt. Medan i en agil projekt gör man det hela tiden istället, kontinuerligt. Det är samma typ av värde som att köra agilt här som det är att generellt bygga saker. Att man har möjlighet att anpassa, möjlighet att följa upp, och man har mycket större sannolikhet att nå ett bra resultat. Och då menar jag bra i form av det

		kunderna vill ha, det som beställaren vill ha. Inte vad de trodde de ville ha för sex månader sedan eller ett år sedan.
60	D	Något som jag inte hann ta upp, i era projekt, vilka risker brukar framstå oftast?
61	I3	Det är integrationer till andra. Så fort det är något utanför teamets makt, så är det en risk som dyker upp. Och det är just för att det är så svårt att göra något åt det. Så ju fler integrationer desto större risker oftast. Det är nog det vanligaste. Sedan är det en del sådana standard risker som folk alltid kommer på när man tänker efter. Folk kan bli sjuka och nyckel personen försvinner och servrarna går i sönder. Det är svårare att komma på de riktiga riskerna och de ibland kan det också hända att man inte tänker tillräckligt mycket affärsorienterat, utan detta är en risk utifrån affärsperspektivet liksom. Det svarar också på frågan om vilka risker som inte hittas.
62	D	Det brukar också ofta vara problem med kundinverkan, att man inte får kunden att involvera sig lika mycket som man hade önskat. Hur brukar ni hantera det?
63	I3	Den är en risk också som finns hela tiden. Och vi brukar mitigera den genom att ha en produktägar proxy, en kund representant i teamet då. Vi anpassar i efterhand om vi ska ha med kunden i varje daily standup eller ej. Vi har tagit bort det i vissa fall för vi kände att det inte behövdes. Men vi provar, och sedan ändrar vi.
64	D	Det var ungefär mina frågor. Finns det något du vill tilläga, något som du inte har hunnit ta upp innan vi avslutar?
65	I3	Nej, jag tycker risk passar bra i agila projekt. Det är ingen motsättning att jobba med risker och att jobba agilt, inget alls. Utan snarare det ingår i det agila projektet att jobba med risk. Bara inspect and adapt, det är riskhantering. Fungerar det här eller inte? Nä, vi skruvar på detta här.
66	D	Precis, det är också en del av agila projekt att man anpassar hela tiden projekten. Det är därför vissa anser att man inte behöver sådana extra processer där man måste hela tiden identifiera risker. Utan det händer automatiskt när man jobbar agilt.
67	I3	Ja men mycket så är det, absolut. Så det är inget man måste ha. Men vi experimenterar. I vissa projekt provar vi att ha en risk burndown för att se om det ger något värde. Precis som vi provar att ha en vanlig sprint burndown i vissa projekt då. Och ibland så tar vi bara bort det. Så det gäller bara att kolla liksom, prova saker och se tillför detta något värde.
68	D	Då tackar jag så mycket!

Interview 4

Row	Speaker	Answer
1	I4	Då får börja med kanske att säga att riskhantering är något som vi arbetar med normalt. Jag personligen har inte såhär att detta är det sättet som vi löser det på, eller som företaget löser det på. Vi har ingen konsensus för hur alla projektledare inom företaget ska hantera risker utan det är lite mer upp till var och en. För min del är det lite upp till varje projekt vi arbetar med. Så ibland har jag inte, såhär och såhär gör jag alltid. Det hade varit väldigt bra, men oftast blir det att det är beroende av vilken typ av projekt och hur uppsättningen ser ut i projektet. Hur kunden ser ut i projektet.
2	D	Vilka typer av projekt brukar ni jobba med?
3	I4	Framförallt, det jag jobbar med är - det är faktiskt ett rätt så bra poäng med hela riskhanteringssystemet, vi arbetar med vår Core produkt. Det betyder att när vi gör en leverans av vår produkt så är det extremt kort utvecklingstid. Vi tar ju en grafisk profil och sen applicerar det på Core produkten. Och sedan är det ett antal möten som man behöver för att sätta upp sin infrastruktur och så för att allting ska finnas på plats. Och sedan gör vi en installation. Och det är det hela tekniska leveransen som görs. Det kan göras på två veckors tid i princip. Så det är mer möten som brukar tar längre tid. Men i praktiken skulle det kunna gå väldigt snabbt eftersom många av dem projekten jag sitter i har inte fokus på riskhantering i det tekniska, utan riskhantering mer i form av kommer innehåll kunna läggas till. Liksom mer mjukare värden. Ofta, är det mer upp till kunden att se över vilka risker har vi nu. Att vi inte försöker att gå in på att säga att ni ska göra såhär i er implementation av ert intranät för att det är upp till var och en och då måste dem köpa det som ett extra tillval för att det kräver så fruktansvärt mycket tid om jag ska gå in och säga till en kund; här har du hela implementationen. Ni ska fylla på med denna information, ni ska göra dem här workshopsen, ni ska föra allt detta och vara med och hjälpa till några, så hade det varit väldigt krävande. Och det är inte så många kunder som känner att de har det behovet, utan de gör det själva och då får de hantera dem riskerna själva också. Långt svar!
4	D	Det är bra med långa svar! Men för att få en bättre bild på era projekt, hur många brukar jobba med dessa projekt?
5	I4	Ja, precis! Förhållande. Detta var en typ av projekt. Sedan har vi andra typer av projekt också. Jag är projekt ledare i ett annat projekt som har pågått i två och ett halvt år där vi har kört agilt enligt Scrum. Väldigt litet utvecklingsteam. Det är det ofta här på företaget, är det oftast små

		utvecklingsteam, 3-4-5 personer är väl det största, 6. Men annars är det mindre 2 till 4 som brukar jobba. Men precis, det är en annan typ av projekt, och då är det knepigt utvecklingsprojekt och då är det mycket mer att vi får hålla reda på risker.
6	D	Och för att få lite förståelse för hur ni jobbar agilt, kan du beskriva lite hur era agila processer ser ut?
7	I4	Ja, absolut. Vi kör enligt Scrum, men sedan samtidigt så är rollerna lite flytande också. Egentligen ska kunden vara mycket mer delaktig och vara den som kravställer. För just detta projekt förlitar de sig på att vi skapar kraven och de godkänner kraven i princip. Det skapar lite mer arbete för oss, men det har blivit så för att de har varit lite osäkra i början. Hur det skulle byggas upp, och vi kom med förslag. Sedan har det bara rullat. Men vi kör tre veckors sprintar, standups varje dag. En inledande sprint möte där vi sätter sprint innehållet och ser till att allt är estimerat så att vi kan hantera det. Estimeringsmöten internt och framförallt inom projektgruppen, och sedan tar vi det när vi har estimerat så kan vi ta det vidare med kunderna. Och sedan kör vi demos också var tredje vecka. Dem har även en referensgrupp av en större skala av människor så att de har varit med i nionde veckan, så i vår tredje sprint. En stor demo. Sedan är det en liten offentlig verksamhet, så där är väldigt många involverade och väldigt många avstämningar. Men utveckling sker enligt Scrum, men sedan är det mycket kringarbete också som inte riktigt tillhör, som inte är en del av vanliga Scrum.
8	D	Vilka delar är inte en del av Scrum? Varför följer ni inte Scrum hundra procent, utan implementerar era egna delar?
9	I4	Det är för att Scrum är väldigt agilt och man har väldigt bra koll på var som händer just nu, men med en offentlig verksamhet måste man hålla koll på vad som händer om ett och ett halvt år och då är det litegrann svårare att göra det bara med Scrum, utan då måste man ha andra verktyg. Man måste göra längre planeringar som inte bara har att göra med utveckling utan som har att göra med hur man ska hantera support och förvaltning, dokumentation. Liksom allting som är runtomkring, kunskapsöverföring. Så det är väldigt mycket möten som är utöver. Det är ju vilket projekt som helst, det är mycket möten. Det är mycket som behöver diskuteras och bestämmas.
10	D	Följer ni en liknande process i varje projekt, eller skiljer det sig mycket från projekt till projekt?
11	I4	Den agila processen brukar vara hyfsat inline. Sedan skiljer det sig från kund till kund, hur mycket av det här extra planeringsmöten och arbetet.

		Men denna grund processen kör vi antingen Scrum eller vår egen variant av agil projekt som är väldigt lik Scrum.
12	D	Du nämnde att ni inte har en fast riskhanteringsprocess, men hur brukar ni hantera risker i era projekt? Hur ser er riskhanteringsprocess ut?
13	I4	Då är det någon slags verktyg först och främst för att hålla koll på riskerna, så då har vi en SharePoint lista eller någon lista i något system där man kan definiera vad är riskerna, vad är sannolikheten, vad är konsekvensen och vad är åtgärderna. Det är denna väldigt grundläggande risk uppgifterna som man måste kunna lägga in, och vem som är ansvarig. Till exempel i detta långa projektet, projekt X, där använder vi Canea framework. Där finns det en dedikerad riskmodul som man bara precis skapar riskerna och delar ut det till människor. Sedan kör vi möten varannan vecka där vi tittar igenom riskerna, ser vad som har förändrats och om det finns någon ytterligare åtgärd som behöver göras eller fördelas till någon annan person, eller risker som behöver läggas till. Så vi går igenom listan uppifrån ner och ser. Och de riskerna som vi hanterar är inte direkt kopplade till Scrum, utan de är risker som är kopplade till projektet. De är kopplade till projektet. De riskerna som kommer upp där är; kommer den andra leverantören kunna fixa API:et som vi måste ha för att komma vidare, eller kommer användarkontona för administratörerna hinna skapas upp i tid, eller vem är ansvarig för det? Så det är inte så många risker i utvecklingsarbetet som hanteras, utan igen är det detta kringarbete, vad behöver göras för att systemet ska fungera så som vi hade tänkt för att dem som sitter med utvecklingen ska ha de verktygen som dem behöver för att kunna göra sitt arbete.
14	D	Den allmänna risken, är det just att externa parter kan orsaka risker?
15	I4	Just i det projektet är det mycket integrationer, och då är det en stor del ja. Jag kan inte komma på några risker som brukar vara med i varje projekt, men det brukar alltid vara någon infrastrukturens problematik. Det är certifikat som behöver finnas på plats, SSL certifikat. Eller om något behöver sättas upp. Jag vet inte, jag kan inte komma på något problem som brukar alltid finnas med.
16	D	Du sa att i detta system där ni registrerar era risker, ni delar ut dessa risker till individer. Så det finns någon som är ansvarig för varje risk. För litteraturen benämner att det är en allmän risk att i agila projekt tar inget ansvar.
17	I4	Ja, det måste man, annars tappar man spårbarhet, ingenting händer om man inte delar ut det. De riskerna som hanteras där är oftast inte kopplade till det rena utvecklingsarbetet. Där har vi inte så mycket riskhantering.

		För att om det kommer upp något som verkar vara en risk så är vi rätt så snabba i det dagliga arbetet.
18	D	I sådana fall hanterar ni det informellt?
19	I4	Ja, det kan vara informellt eller det kan vara formellt också. Om det kommer upp något under tiden vi för utvecklingsarbetet så lyfts det och sedan gör vi en kontroll, vad är läget? Är det så att där är en risk som vi får reda på att där är väldigt långa svarstider på ett källsystem och vi vet inte varför, så gör vi kanske först en undersökning så visar det sig att det finns något som inte stämmer. Då lyfts det problemet från utvecklingsteamet in till den övergripande styrgruppen där riskerna hanteras. Så det blir på det sättet dem riskerna hanteras. Dem lyfts ut till den övergripande projektledargruppen.
20	D	Och sedan nämnde du ett riskregister för att skriva in och registrera riskerna. Har ni andra artefakter eller verktyg som ni använder? För att övervaka riskerna. Som till exempel en risk burndown chart, eller Kanban?
21	I4	Nej, ingen jag kan komma på utan det brukar vara basic. Som listor, där man absolut kan köra lite Kanban genom att man sätter att de är pågående eller att de är done eller så. Men i arbetet är det ganska enkla verktyg, list objekt liksom. De kan kanske presenteras på ett lite trevligare sätt än bara excel. Excel brukar vi inte använda så ofta faktiskt, utan vi använder ofta någon form av system om det är SharePoint eller Canea osv.
22	D	Du nämnde att ni registrerar risker i registren så brukar ni ha en åtgärd associerad till en risk. Ni kollar kanske på vad sannolikheten och impacten är, och sedan har ni kanske en exposure rate. Och beroende på exposure raten, har ni en fix lösning? Hur ser det ut? Har ni standarda lösningar till standarda problem?
23	I4	Nej, det har jag svårt att föreställa mig om hur vi skulle kunna ha. Det skulle kanske i så fall vara om man har någon annan typ av projekt eller verksamhet. Det känns lite som att här dem riskerna vi får är väldigt individuella för varje projekt. Ibland är det problem med, om vi tar detta certifikat problemet, alltså att riskerna inte framkommer i tid så absolut kunde vi ha en mall, såhär ska man göra i det här fallet. Men samtidigt är det ganska givet vad man behöver göra, det är att kunden behöver jaga ut och jaga tag på certifikat. Så ofta är det inte svårare åtgärder, utan det är bara ledtider som gör att man inte kan göra som man har tänkt. Nej, jag tror inte vi använder några standardiserade lösningar.

24	D	När ni kollar på er nuvarande riskhantering process, upplever du att den är hjälpsam? Eller är det redundant i och med ni jobbar agilt? Eller är det bristfällig?
25	I4	Nej, det är väldigt användbar. Alltså när det är långa projekt och många variabler, då behöver man ha någon sorts riskhantering. För det är vissa saker som man inte kan hantera bara i iterationerna. Det kan vara en risk som man behöver vara uppmärksam i månader eller till och med år. Vi har haft en risk som är två år gammal som ökar upp och ner i sannolikheten baserad på hur läget är just då. Det har jag svårt att se hur man skulle hantera bra i Scrum utan en specifik risk management system. Så väldigt användbart.
26	D	Anser du att man behöver en sådan riskhanteringsprocess när man jobbar agilt, eller klarar man sig utan? För att bygga på det du nämnde just nu.
27	I4	Man skulle kanske kunna klara sig utan om det är ett kort projekt och man har väldigt bra kommunikation där det inte är så mycket variabler runt omkring, men jag tror att det är rätt så sällan det händer. Jag tror i de flesta projekten behöver man ha någon slags riskhantering för att vara medveten om möjliga problem.
28	D	Har ni haft ett projekt tidigare där ni inte har använt riskhantering?
29	I4	Det har vi nog haft, men då har det varit någon annan typ av riskhantering. Flummigt svar! Om vi inte har haft en dedikerad risklista, så har vi istället haft en aktivitetslitsa. Där följer man såhär att, denna person måste göra detta. Och det är denna person ansvarig för. Om där är något som behöver hanteras, i princip om där är en risk för något skulle kunna skapa problem senare, så lägger man in den i aktivitetslitsan och så hanterar man det där igenom istället. Även om man inte säger att vi kör på en dedikerad risk modul eller system så tror jag att per automatik hanterar man risker på ett eller annat sätt ändå.
30	D	Anser du att när man har riskhantering integrerat i agila processerna, att denna riskhanteringsprocess brister mot de agila principerna?
31	I4	Nej, det tycker jag inte. Så som jag ser det så stöttar den. Jag ser ingen krock på det sättet.
32	D	Till vilken utsträckning tycker du att riskhantering påverkar era projekt? Till vilken utsträckning blir det bättre eller sämre?
33	I4	Det är svårt att konkretisera, men det är klart att det är bra om där är ett problem eller et möjligt problem att det lyfts och att man inte behöver tänka på i det dagliga arbetet, utan i det dagliga arbetet fokuserar man på det man har i den prioriterade backlogen. Och inte tänka på vad som

		behöver göras eller hanteras om tre veckor när risker blir aktuell igen. Så jag skulle säga att den har en stor fördel.
34	D	Har du tidigare jobbat med traditionella metoder, så som vattenfallsprojekt?
35	I4	Inte så mycket tyvärr, nej.
36	D	Har du någon åsikt på hur riskhantering i agila projekt skiljer sig från riskhantering i traditionella projekt? Om man behöver det mer i agila metoder eller i traditionella metoder?
37	I4	Jag har ingen praktisk erfarenhet av det, så jag vågar inte riktigt säga. Jag vet inte, spontant skulle jag säga att vattenfall skulle kräva lite mer riskhantering, man har inte samma möjlighet att svara snabba förändringar. Det är nog inget du ska citera mig på, för jag är inte säker på det.
38	D	Och sista, for the record, vad är din roll i företaget?
39	I4	Jag är projektledare. Började på företaget som utvecklare och har varit projektledare de senaste tre och ett halvt åren.
40	D	Okej, och innan vi avslutar ville jag ta upp igen detta med att ni inte har haft projekt där ni inte har haft riskhantering direkt. Men återigen, under vilka omständigheter anser du att man inte skulle behöva riskhantering? Finns det några faktorer som avgör om man måste ha riskhantering.
41	I4	Jag vet inte, ett utopiskt projekt där man vet att man har allting som behövs på förväg och alla är med noterna liksom. Till viss del dessa projekt som jag började berätta, där det kan vara rena tekniska leveranser som ska vara klara i två veckor, jag vet inte ens om jag skulle kalla det för Scrum eller agila projekt. Vi kör dem i sprintar, men det är fortfarande så pass korta projekt. Vi försöker att köra dem enligt en agil metod, men jag vet inte om man kan typiskt kalla det för Scrum eller agilt. Där ifall skulle jag kunna säga att man inte behöver så jättemycket riskhantering eftersom de är så pass korta, vi har gjort så pass många förut att vi vet i förväg vilka krav vi behöver ställa på kunden, vilken information vi behöver få i på förväg. Det kanske det är, om man har gjort något tillräckligt många gånger förut så vet man vad som behövs och då behöver man kanske inte i samma utsträckning riskhantering. Men sedan är det alltid kunder och projekt som inte kör på samma bana, som alla andra gör. Där det uppstår konstiga situationer. Där får man nog vara beredd att använda sig av riskhantering i sådana fall.
42	D	Ja, då har jag ställt alla mina frågor! Finns det något du skulle vilja tilläga innan vi avslutar?

43	I4	Nej, inget vad jag kan komma på.
44	D	Då tackar jag för mig!

Interview 5

Row	Speaker	Answer
1	D	Vad är din roll i företaget?
2	I5	Ja, min roll är projektledare och kravställare.
3	D	Hur länge har du varit projektledare?
4	I5	Ja, det är väl fyra år kanske. Man har glidit in det mer och mer. Jag började som utvecklare från början. Och sedan har man jobbat som arkitekt, och sedan har man suttit där. Så det har smugit mer och mer. Men det är först nu när jag började jobba här som jag är enbart projektledare.
5	D	Och du är projektledare främst för agila projekt då?
6	I5	Ja, det är det enda som jag driver.
7	D	Har ni traditionella projekt också här hos företaget?
8	I5	Jag skulle säga att vi inte har, men det har man ju. Det är lite mer grejen att vattenfall som man brukar säga, det är ofta det som man brukar hamna i. Där kan det finnas olika orsaker. En orsak är ofta att man kan vara tvingad till att gå den vägen för att kunden inte är mogen nog för att köra agilt. Eller kan det vara att man hamnar i en situation så att om man har en offentlig upphandling så kan man oftast spara en förfrågan, ett avrop, och då blir det igen att man ska göra ett estimat för att vinna upphandlingen. Det handlar också lite om problem som egentligen agila projekt inte tar höjd på samma sätt. Eller de attackerar på annat håll, när man jobbar agilt.
9	D	Det är alltså kunden som bestämmer i slutändan.
10	I5	Åter, vill kunden veta vad en kostnad blir, och det blir även grejen att om man skulle säga Scrum, ja, det vet vi inte men vi vänder mycket på det beroende på hur mycket man vill investera. Och sedan kan man säga, okej, hur mycket kommer vi få ut från detta? Och det får man se utifrån de olika premisserna, hur teamet är, och vad man har för saker man ska jobba med. Det brukar inte funka i början, men då vill man ofta såhär, att vi vill ha detta gjort i oberoende av hur mycket det kommer kosta. Senare vrider man in det mot agilt och ofta så inser man ju att både från kundens och vårt håll att det går ju inte, det man tyckte var bästa lösningen från början förändras med tiden. Osäkerheten blir mindre, ju längre in man

		kommer ju mer man vet om saker och ting och då förändras bilden. Då kan man fatta annorlunda beslut än vad vi gjorde från början.
11	D	Det är också det med agilt, att man vill anpassa sig till förändringar. Om vi går vidare till hur ni jobbar agilt? Om det är Scrum eller om ni har era egna processer till det?
12	I5	Vi har en projektmodell som egentligen bygger på Scrum genom att vi har med oss projekt som bygger från initiering av projekt, alltså in försäljning av ett projekt. Och sedan driva igenom till att man börjar driva projektet, att det är godkänt och att det blir ett projekt. Och sedan när projektet är avslutat, då tas det över till förvaltning. Där är det större, men i den här projektmodellen då så ligger Scrum med sprintar och backlog och iterativ och inkrementellt utveckling.
13	D	Ni har alltså en längre IT projektlivscykel, men det är delad i sprintar.
14	I5	Ja, det blir ju det för att, en del är att man är i själva projektet när det verkligen är live, när man driver det. Men där har man ju också in försäljning innan, också en del kostnad som man måste ta med. Det för att säljarna driver sitt sälj tills man har lyckats med affären. Och då har man ju, vid det tillfället, redan en hum om hur stor projektet är. Det är ofta så att vi går in - ja, det här projektet är estimerat att det ska vara på tusen timmar eller mer eller mindre. Och ett grovt backlog eller kravspec, dvs. vad vill man uppnå? Om det är så att man uppgraderar eller om man vill implementera ett nytt system, vad är det då som det skulle omfatta? Med det bryts det ned till en backlog som är detaljerad för att kunna driva igenom hela projektet. Och sedan efter det så kommer man ha avslutat det, då tar vi emot och tar hand om förvaltning av det som vi har implementerat. Då går vi över till ett nytt projekt där man kör ett förvaltningsprojekt. Och det pågår länge, vi har haft kunder som vi har jobbat med i tolv år. Så det är mycket längre, men det är fortfarande i sig ett projekt.
15	D	Underhållning av systemet brukar vara det som dröjer längst. Men brukar ni följa dessa processer för varje projekt?
16	I5	Nej, de skiljer sig nog mellan - det är skillnad mellan i vilket område. Om det är CRM eller custom solution, eller BI. Så tror jag att det är lite annorlunda i hur man driver det. Vi försöker att hålla en gemensam, men man får väl ta det till så lång det går. Huvudsaken är att driva så att det ska vara, och det är ett mål som vi har här internt också, att projekt ska se lika ut i struktur och när man jobbar för att kunna växla mellan personer i projektet. Om man inte är i ett projekt så ska det inte vara något helt nytt på det viset. Man ska fortfarande vara van vid hur det drivs. Det är ju för att, ja men som sagt vi driver det enligt Scrum och det är Scrumish. Så det är ofta det, att man behåller backloggen och man jobbar efter sprintar och sådär. Men sedan kan det också vara, i själva stommen men det finns saker is Scrum som tas upp som är mycket svårare att hålla uppe. Ett helt team som får vara ostört och även att jobba med sprintar och kunna efter

		varje sprint att utvärdera och förändra... Dem delarna kan vara lite svårare, men vissa kunder är bättre på det och mer mogna för att jobba på det sättet. Andra kunder är inte det.
17	D	Som du sa, det blir kunden i slutändan som bestämmer om man jobbar agilt eller inte.
18	I5	Ja, det blir ju det. Det blir ju inte bra ändå att gå in och jobba agilt med en kund som inte är agilt i sitt tankesätt för att det krockar mot verksamheten och då faller egentligen det agila för att det bygger egentligen på att alla är på tåget där. Det är ju klassiska med traditionellt vattenfalls projekt om man får kalla dem för det, så kan det vara så att dem har ingen product owner, och har man inte en dedikerad product owner som fattar beslut, då går det inte att driva en Scrum projekt, för då haltar man redan där. Det är ledtiden från att man har en fråga till att man har fått ett svarbeslut. Det tar för lång tid, det går inte att driva agilt.
19	D	Brukar det vara den främsta risken som ni har med era projekt? Att kunden inte är tillräckligt agilt? Eller vilka risker är mest förekommande?
20	I5	Ja, en risk är jag kan tänka mig är just product owner, alltså ägare, som är aktiv. Det är stora risken. För om man ska driva agilt så hjälper det inte att man har någon som är ansvarig. Den personen måste aktivt vara med i projektet. Det betyder att kunden i sig måste ha uppsatta tid, kostnad för någon internt i sitt företag som jobbar både med projektet tillsammans. Har man inte det, då blir det mer såhär att man måste ha saker och ting kramas från början, det måste vara dokumenterat, och sedan jämföras. Men alltså den här aktiva rollen, det har jag sett, det tycker jag kan vara svårast att få med. Och det är egentligen första nyckeln, faktorer för att kunna hitta rätt från början. Sedan därefter kommer det product owner, och det är verkligen en sådan roll, och det har man oftast - man tror att agila är iterativt och inkrementellt, men det är inte det som agilt egentligen, dess huvudsyfte. Dess huvudsyfte är att man ska, när man tittar på Scrum och Kanban och sådana, att man har värddrivet och inte process drivet. Det är därför man har vattenfall och RUP och alla dessa, dem har kanske viktigare processer som hakar in på varandra. Tittar man på projektet så ser det ut som allt går jättebra enligt processen. Men tittar man på vilket värde det genererar så är de inte bra. Och Scrum vänder på det istället och säger att vi väljer att se på värdet som genereras, och inte processen. Processen finns där, men det är inte det som är själva backbonen i Scrum. Det är att det ska hela tiden generera ett värde för product owner.
21	D	Ja, man definierar succé på olika sätt i dessa metoder.
22	I5	Precis, och det man ofta mäter ändå, i vilket projekt man än driver med, så är det till slutändan nog det man mäter. Och kunden blir besviken och säger att nämen det här var inte som vi hade tänkt oss. Nej, men om man hade tagit det från början så hade vi kanske kunnat säga att vi ska lägga ner så här mycket energi för att kunna - det är lite det, vad man investerar i. Jag har haft några kunder som har mer tid i uppdraget som bara vill ha det gjort. Man har kanske inte kunskapen i området, om man förlitar sig

		på att konsulterna löser det. Men så funkar det inte riktigt. Det krävs verkligen att vi kan ju vår del, men det vi inte kan är ju själva business value för företaget. Det är bara företaget som är bäst på det. Vi kommer inte kunna komma in på den nivån att vi vet exakt vad det är för business value som är viktigast, utan det ligger på ledningen och allt annat. Och det måste då in i projekten. Då är ju product ownern som är nyckeln nu.
23	D	Innan vi går vidare till risk management, en annan bakgrundsfråga, du nämnde lite om CRM och så, men vilka typer av projekt brukar ni jobba med främst?
24	I5	Här har vi CRM, vi har custom solution, vi har business intelligence, det är väl dem som jag vet är de stora. Vi är ganska spridda på ett stort område så ju. Och sedan har vi, vi som företaget har bestämt att vi vill både ha teknik, att kunna tech men även strategi, att kunna jobba på den nivån. Och jag sitter med CRM nu.
25	D	Och för att få lite bättre förståelse för projekten, hur stora brukar de vara? Hur många brukar jobba med dessa projekt?
26	I5	Det kan vara från en person upp till fem personer och uppåt. Ibland kan man ha ett projekt som börjar smått, växer till stort och sen blir smått igen. Men det som jag sitter med nu, det är väl bland vårt största projekt. Storleksmässigt så är vi väl totalt fjorton där.
27	D	Sitter ni distribuerat?
28	I5	Vi sitter i samma rum. Det behövs, det här att kunna kommunicera och inte bli hindrad av teknik, det är mycket lättare att man hamnar i tekniska verktyg för att kunna kommunicera, men Scrum vill ju hela tiden att man har standup, och daily Scrum och så. Det är mer för att det är där man ser, det går snabbare än att man ska kommunicera på ett annat sätt. Men de kräver också tid. Det kan inte vara större än en viss mängd. Nio är väl det som man säger är gränsen och vi har ju gått över till att köra Kanban nu, lite för att kunna komma förbi det att vi är för många för att köra i ett Scrum team och vi har svårt för att dela det som två Scrum team för det är svårt att veta var vi ska dela någonstans.
29	D	Om vi går vidare till riskhantering, kan du beskriva i detalj hur er riskhanteringsprocess ser ut när ni jobbar agilt? Om ni har som traditionella riskhanteringsprocessen att man definierar risker, planerar för dem, gör något åt dem, övervakar dem. Eller hur ser det ut för er?
30	I5	Oftast gör man riskkalkyler och sånt i början, project definition och så, vad är risken. Hur kan man lägga upp risk? Hur allmänna saker inte finns plats, att man inte har resurser eller så. Om man väl kommer in i själva - när man jobbar i Scrum - en risk som alltid finns är att man ofta vill ha sin prognos eller forecast. Alltså vart lägger vi tiden? Kommer vi hålla deadlinen som man har framför sig? Det finns även risk på teknisk nivå som man kan tänka att - hur vet vi arkitekturen blir som den är? Det är också en risk att man oftast - i Scrum tar man inkrementellt och iterativt, det är enkelt att säga att man jobbar men det är väldigt svårt att jobba efter det. För att man har en arkitekt, man bygger upp systemet, man

		<p>skissar upp det och sedan därefter börjar man ta sig igenom och börja implementera. Men i Scrum så börjar man med att skissa något, men det finns många områden som inte är riktigt satta. Man vet ungefär vad man gör, men det är inte satt till detaljerad nivå. Om ett halvår framåt kommer vi fortfarande inte veta exakt hur vi kommer vara där. Det har kanske skett, kanske är så att det har skett saker som gör att det vi tycker är bästa lösningen idag inte är bästa lösningen om ett halvår utifrån att - ofta så finns det flera faktorer. Går man in i ett projekt så är alla osäkra. Både vi och kunden. Man kan vara säker på att man kanske vill att ha en viss sak. Det behöver inte betyda att det egentligen är - om kunden har ett problem, så tror vi att lösningen är att man gör en uppgradering. Och så går man in i projektet och man blir mer och mer medveten om kontexten osv. Så kanske man märker att lösningen på problemet inte är en uppgradering utan det kanske är en verksamhetslösning. Och vad gör man då åt risken? Ska man ändra riktning eller ska man köra vidare och köra en uppgradering? Och enligt traditionella projekt, lite såhär, så kan jag säga att det blir en uppgradering. Och så mäter man och så ser man att det blev ju inte bättre. Nä, uppgraderingen löste inte problemet om det var så att det var verksamheten som inte kan konversera sig på. Eller processer som inte fungerar. Uppgraderingen löste inte dem processerna, och då finns fortfarande problemet kvar fast i en ny version av det man ville uppgradera. Så det är ju risken så här, och agilt bygger mycket - det är därför de har sprintar. Man säger så här att - man kan inte ha mer än en sprint per månad. Det man kommer fram till att - i längden så börjar man famla för mycket i mörker av vad man egentligen vet. Man bygger lite mer såhär just in time. Vad ska vi göra utifrån den kontexten vi sitter i, utifrån det vi vet och så går man efter det. Därför kör man då sprintarna, kör kortare för att hela tiden, efter varje sprint, utvärderas - är vi på rätt väg? Har någonting förändrats? Måste vi fatta några andra beslut än de som vi trodde för en sprint sedan, för en månad sedan? Hela tiden titta tillbaka där. Och det är ett sätt att hantera denna risk, för antingen satsar man allt i ett bräda, och så kör man igång det. Man har investerat allting, men det är verkligen - prickarna går det här att inventera? Kommer vi i testfasen att verifiera ut - men det som agilt har sett är att kostnaden för att göra en ändring blir dyrare ju längre det har gått i ett projekt. Förändring i början är mindre kostsamt än att göra ändringen och vända tillbaka när man har kommit in i en testperiod, och gå tillbaka och ändra. Buggar kostar mer än att ändra saker under utveckling. Och det är också en risk som då agilt och Scrum och Kanban osv löser. Scrum säger - okej, istället för att investera så mycket pengar i och med risken och kostnad blir så hög så man kan inte ändra, man kommer till återvändningsgränder. Det är att - det har gått så lång och det har investerats så mycket tid, och pengar, så ändringen blir nästan inte - går inte att göra. Så vill man ha det mycket tidigare.</p>
31	I5	<p>Och det är lite det också som - när vi sitter med kunden som jag jobbar med just nu, vi kör devops med continuous build och continuous deploy till produktion. Det är samma sak där, att man vill hela tiden komma</p>

		<p>längre fram - det här att utvärdera och göra förändringar så kommer man mycket närmare i horisonten. För att kostnaden är lägre när man har just in time än att man ska köra genom en process och sedan efteråt så verifierar man och utvärderar man och så går man tillbaks så här igen. Man vill ha små iterativa steg, och så inkrementellt bygga på. Och det är också det som om man säger risk där, det är väl med inkrementella så eliminerar man också risker med att man inte vill skapa saker som inte returnerar något värde. Alltså värde driven utifrån return of investment. Om man har till exempel suttit i CMS system och även CRM man kan ha en uppfattning om hur en funktion ska fungera osv och sedan så bygger man den här funktionen och så finns lite det här - 80/20 regeln som oftast finns och gäller nästan alla saker man gör. Och då märker man att när man har då gått efter den här specen som man har tänkt sig - nä men det här är bra, vi behöver det och det. Och sedan släpper man ut det och så märker man att - nämen användarna använder ju inte allt. De använder en viss del, men mycket av det som vi har lagt ner tid och pengar på används inte. Och då - okej, om vi kör inkrementellt istället och MVP, att vi bygger så vi känner så att nu kan vi tillgodose en funktion som går att använda, ut med den, användarna får testa eller använda den, kolla på feedback direkt mot användaren, vad är det mer de behöver? Det här! Bygger man på, och bygger på. Så man kan inte gissa hur en användare arbetar eller hur man vill ha det, det är egentligen användaren som har det svaret. Det är också ett sätt att hantera risker, att inte bygga saker som kommer ut och så är 20 procent av funktioner används av 20 procent av användarna. Det blir det här att man vill få ner den risken. Det är också en sådan svårt - som man kanske inte kan göra i vattenfallsmodeller, för där har man ju allting. Där har vi kartan framför oss, blueprinten, och sedan så kör vi igång. Och så kör man igång alla processerna och ser man och utvärderar utifrån den kartan man har gjort från början och lyckat eller lyckat, det höll ju tiden. Har vi uppfyllt allt som vi sa vi skulle uppfyllt från början? Då har vi sett många gånger att - fast man har hållit tiden, fast man har uppfyllt allt som man har sagt att man skulle uppfylla, så är kunden eller användaren inte nöjd. Då har man sett att - men vad hände på vägen? Jo, saker och ting förändras, kontexten förändras, men kartan har inte förändrats. Så den man implementerar gäller inte idag. Och den risken är ju det som Scrum och agilt hela tiden jobbar med - att man inte ska hamna i det spåret utan jobba just in time, MVP alltså Minimum Viable Product, för att hela tiden, och det är ett sätt att eliminera risken att inte over-do it.</p>
32	I5	<p>Men det är lätt att man tar en enkel funktion som att - om man jobbar i slutet av projekt- om man jobbar med appar, så har man haft ett krav att appen ska kunna fungera offline. För man säger att det är bra. Det är jättebra om man kan använda det utan nät. Och så har man gjort mycket jobb för att kunna få den att fungera offline, men sedan börjar man märka att funktionen i sig kanske fungerar, men den har ingen nytta offline i helheten eller i appen. Och då kan man säga att man har lagt ner mycket på att göra ett krav som egentligen inte kommer användas. Då har ni ju</p>

		<p>plöjt ner mycket risk och risken av att få mycket pengar kanske på det sättet. Det kan även vara såhär att man ska ha ett bakåtkompatibla litet. Ibland kan man vara för hård på det, men tittar man på det så kan man se att - okej, men i den kundkretsen vi har, om vi går tillbaks och har den här - det ska fungera ett antal versioner tillbaka så - de här kunderna som använder det här är så pass få så det kostar så mycket för att göra de kunderna nöjda. Så det är också en risk, och det är därför det är viktigt att ha med den här product owner och ledningsgruppen i företaget - för det är de som har den här business value. Det är en sak att tekniskt implementera det, men det har ett business value bakom. Och då måste man se att funktionen kanske är jättebra, men den har inget värde i detta sammanhang. I ett annat sammanhang, för någon annan så har det kanske jättestort värde, men just här - jag har en stor kund som jag har suttit kommunala, och offentliga verksamhet, och privata. Och när man jobbar offentligt så har de lagar och krav på sig som inte de privata har. Gör man till exempel en hemsida så ska den vara tillgänglighetsanpassad för alla medborgare i en kommun. Det är en kostnad, men det måste man göra för att lagen säger att alla ska kunna ta till sig informationen för det är samhällsinformation som är viktigt, det ska nå ut till alla. På ett privat företag så släpper man kanske ut en produkt om man har ett företag som släpper ut en nischad produkt. Då har man inte de kraven för man känner såhär att, nää vi är inte ute efter att sälja produkter, det är inte så att alla ska ha tillgång till det. Men de som vi vill rikta oss till, de ska kunna nå den. Och då går man efter det hållet, men det kan också vara så att - det skiljer sig mycket, och det är också en risk där. Och där hittar man i - som jag tycker med agilt som - det tar hela tiden risken, det förändras och även att det jobbar med inkrementellt och iterativt. Det är ett sätt att hantera risken i det stora.</p>
33	D	<p>Från vad jag har förstått så hanterar ni risker genom att jobba agilt. Som du sa, att jobba inkrementellt och iterativt. Men utöver det, har ni några, till exempel som traditionella riskhanteringsprocesser, att man har en risk register, att ni definierar risker i början av ett projekt eller liknande explicita riskhanteringsprocesser?</p>
34	I5	<p>Näe, inte som jag jobbar med. Det finns ju - oftast ligger det som du sa i början, att man lägger ner risker - man kollar på vilka risker som finns med det här projektet. Och sedan så har man det i ett gruppmöte. Men de tycker jag är - de ligger lite mer på en högre nivå. Är det en risk till exempel, det kan vara en produkt, som har en risk att vi måste ha det i mål i ett halvår. Vi har en hög risk här, så här måste vårda den, det måste ha släppt inom en viss period för att det kanske ska vara till ett visst ändamål. Och det är en högre risk där. Hur ska man hantera det? Eller att en annan risk vi har haft i vissa projekt som gör att vi till exempel inte har gått in i det - i det här projektet så är där för många externa leverantörer. Att man har en risk - okej, vi är en del av flera stycken som ska samarbeta. Risken där är då att - kan man inte ha en kommunikation mellan, kommunicera med dessa externa leverantörer, så kommer det kunna vara risk för oss att gå vidare i saker och ting, vilket kan vara till</p>

		<p>exempel att om vi ska utveckla något och så är det kanske hostas det här hos en annan leverantör, då är det ju en risk att vi kan bli hindrade och bli blockade av att den leverantören kanske inte kan leverera det man har lovat. Och i sin tur, göra att vi inte kan implementera det som vi har lovat till kunden. Så det är ju en risk där. Och det kan man se med min nuvarande kund, att molntjänst, där har vi Microsoft som är leverantör av molntjänsten. Och det är ju en ganska stabil leverantör i sig plus att vi själva har tillgång till serverna och de sakerna som är i molnet. Då är ett sätt att eliminera risken - det hade varit en annan sak om det hade varit ett annat företag som hade hostat servrar och haft en serverhall och hela den grejen. Då hade ju kunden behövt skapa avtal med dem, och sedan få upp det, och allt som det innebär. Sedan så skulle vi då komma in i bilden och komma åt dessa servrar och kunna jobba mot den leverantören. Att implementera och deploya programvara. Så det är en sådan typ av risk, men det ligger innan själva projektet går igång. Sedan när projektet är väl igång, då brukar det ofta vara - som jag säger, det ingår ofta att hålla deadline, man har alltid att det ska avslutas till en viss tidpunkt. Kostnad, det är alltid en risk. Och att man uppfyller kravet som de hade.</p>
35	I5	<p>Det är också agilt att - kollar man på vattenfall så har man mer såhär att man har fått med sina risker sedan så har man haft diskussioner så att nä men, vi tolkar det så här. Ja men bra, då är vi överens om att vi har tolkat dessa krav rätt. Sedan går man vidare så kollar man och säger såhär att - ja, nu har vi gjort hela listan. Vi har gjort så som ni har sagt, vi har uppfyllt dessa krav som står här, så vi anser oss färdiga. Och sedan om man har tur så sitter kunden också med. Har man otur så sitter man kanske och diskuterar en sak - jamen det står här att det ska vara så. Ja men vi tolkar det så här. Så blir det en tolkningsfråga. Men om man har fortfarande med det här - agilt - det för där har man egentligen en backlog och så ser man att det som ligger i backloggen, det kan vi jobba med. Vi har en viss investerare kostnad, det är egentligen tiden. Om vi är fem pers och så har vi en backlog som säger att vi har tusen timmar. Kostnaden för fem personer, och tusen timmar, sedan vad varje person kostar. Det är vad man har investerat. Och så har man en backlog. Och det är egentligen - Scrum säger såhär att - okej, då behöver vi jobba, och så kör vi på sprintar och sedan i dessa tusen timmar kanske vi får in ett antal sprintar, och så behöver man plocka av backloggen. Men det kan också läggas till saker. Som sagt tidigare, just in time, man tar inkrementellt och iterativt. Det betyder också att det man - som backloggen ser ut från början, kommer inte se ut så när man är mitt i projektet och i slutet av projektet. Saker och ting kan försvinna. Det kanske är något item med en user story som man ser senare att det ligger långt nere, men det börjar komma upp sen om det är aktuell fortfarande. Det som vi trodde var ett krav som vi behövde uppfylla, det behöver vi inte uppfylla nu. Det kan vara så att det har kommit till fem andra som vi inte ens hade tänkt på, och dessa måste vi göra. Och då är det lite så att - man vet inte riktigt vad man kommer få när man är färdig.</p>

36	I5	<p>Om man drar gränsen på tusen timmar, då kan man inte säga - det är svårt att säga så att ni kommer få det som ligger i backloggen. Det kan inte Scrum säga. Scrum kan säga, ja vi kan göra ett estimat. Vi har jobbat nu med två sprintar och nu vet vi vad vi har för velocity, på fyrtio poäng, så tittar man såhär - med den här velocity och som backloggen ser ut så borde vi kunna dra strecket där. Så det kommer vi troligtvis hinna med som det ser ut idag. Men det är det som är lite svårt att komma överens med kunden. Oftast vill kunden gå igenom allt de vill ha. Då det är risk där, risken att veta - och Scrum väljer, de säger såhär att - projekt drar över i vilket fall som helst ju. Man har stora projekt, och sedan har man dragit på sig och så har det dragit över. Man har sett att man ska få in mer saker, och tvärt in med resurser och så har projektet dragit över. Man säger att en stor procent av projekt är misslyckade projekt. Det har varit en ganska hög siffra ett tag. Det har varit det för länge. Nu vet jag inte om Scrum projekt är mycket bättre på den saken. Det är lite såhär, vad är ett misslyckat projekt? Scrum säger lite såhär - väl driven rutin - om det genererar tillräckligt mycket värde som de behöver, tillräckligt mycket värde om man har investerat. Ofta investerar man ett värde och det ska ge mer värde ut. Då är det ett lyckat projekt. Det betyder inte att - projektet kan dra över dessa tusen timmarna. Det kan bli tvåtusentimmar. Men det kanske - om man tittar på ett Scrum projekt så kanske dessa tvåtusentimmar investeringen, ger ändå mer värde tillbaka. Men det är lite den här risken att kunna jobba på det sättet. Det är lite omställning i huvudet om hur man ska göra. Det går ju - på vår nivå är det enkelt att driva det. Men vi håller på med velocity och allting. Går man högre upp i en ledningsgrupp så kan de ha svårare för det. De kan säga - ni har sagt att ni behövde tusen timmar, men nu säger ni att ni vill ha tvåtusentimmar. Så det blir en risk där, hur kommunicerar man fram det?</p>
37	I5	<p>Det som jag har kört är väl att - får man upp en velocity som man kan mäta, så plockar man in något i backloggen, så förändrar det tiden. Men om man tydligen visar - tar man in ett krav, en user story så lägger man till ett antal poäng. Om man vill försöka hålla kvar, stanna vid tidsgränsen, så måste någonting bort. Och det är ett sätt att jobba med risk. Du kan inte bara plocka in. Utan - man har prioritering i backloggen, tanken är att man ska hela tiden prioritera det som ger mest kritisk värde överst, och sedan måste du minska det. Så jobbar du här uppe och kommer ner här, och drar vi gränsen här, så ska man känna att - nja, det hade varit bra om det hade varit med, men vi har ändå fått med så pass mycket så vi fortfarande har fått med det som vi behöver. Så det är ett sätt att - och det är det här att hela tiden inkrementellt anpassa sig efter förändringen i kontexten av projektet som man jobbar med.</p>
38	D	<p>Och brukar det finnas några som tar ansvar för risker? Eller hur står det till för er?</p>
39	I5	<p>Det finns alltid en projektledare ju, har vi ansvar för hela projektet ju. Är man projektledare så har man redan ansvar för att se till hur budgeten ser ut och rapportera till ledningsgrupp och så. Men sedan när vi bygger agilt</p>

		<p>så är det ett ansvar som ligger i teamet. Det är det som är tanken att - teamet är ytterst ansvarig för själva arbetet i sprintar. Så det blir lite den grejen alltså att tänka till och det ställer ju krav också på dem som utvecklar, att de ska kunna gå in och se att en user story ska ha en levererat värde. Finns med den här risken till exempel att man sitter och gör user stories för att man vill förbättra saker men ger inte något värde? Det är ganska farligt, för vem ska betala för det? Man har en kund och sedan måste vi göra lite refactoring här för om man tycker det hade varit bättre om vi gjorde på detta sätt. Men det kanske för med någonting till funktionen. Då kan det vara så att - vi genererar inget värde egentligen, så ska man inte göra det. Däremot så kan man - har man andra regler eller ledord som skulle kunna vara att lämna ifrån sig koden, funktioner - det ska vara lika bra eller bättre. Varje gång du gör det inkrementellt så ska det ju bli bättre. Just att bara slänga in och sedan - nu måste vi göra det här för att guldfila koden. Och det är en risk, att det vi tycker är viktigt inte är viktigt för kunden. Och det blir mycket mer synligt i Scrum.</p>
40	I5	<p>I varje projekt så har man ju ett krav. Så jobbar man på, så händer det en massa saker bredvid men Scrum och Agila projekt, Kanban osv. De vill hela tiden lyfta upp transparency. Alltså att det är synligt, de vi gör, för att kunna se. Men nu jobbar vi på något som inte genererar något värde. Det ska elimineras. Så det är det som är målet. Och det är också ett sätt att hantera risker, att projekt som börjar skeva, att man sitter och gör saker man tror eller tycker är viktigt, men som vissa inte är viktiga om man frågar kunden.</p>
41	D	<p>Och återigen, en liknande fråga, hur ni hanterar risker när ni jobbar agilt. Är det ungefär samma för alla projekt, har ni en gemensam process för riskhantering som implementeras för varje projekt? Till exempel att i början av varje projekt så ska vi definiera risker, göra en lista eller så.</p>
42	I5	<p>Aee, alltså vi har återigen det här projekt definition. Alltså den som lägger risken där. Men sen är det så ju, när man jobbar med projektet så är dels risken uppe på en annan nivå. Där har man stora risken, men när man jobbar inne i sprintarna så är risken egentligen hela tiden att - en risk är att man gör en user story som är beroende av varandra. Den här user storyn måste göras innan den andra user storyn görs osv. Och då har man en hel kedja med user stories, vilket betyder att man inte kan prioritera user stories i en backlog. Det är en risk, det betyder att om inte denna user story görs så kommer inte dessa andra kunna göras. Och helt plötsligt blockerar man ett helt antal funktioner. Så det är ett sätt att, det ska helst vara user stories i en backlog som är isolerade från varandra så mycket som möjligt. Det går inte alltid, men det är det som är målet. Det är också ett sätt att eliminera risk. För att företaget ska kunna prioritera och man ska inte säga såhär att - ja men detta kan du inte göra förrän du gör det och det. Och så sitter man och har bundit in sig i en slags kedja av funktioner som måste göras i en viss ordning. Det finns också risken av att - tänk om den här funktionen inte behövs? Och samma sak om man kollar på det tekniska, hur själva user stories är uppbyggda och delade.</p>

		<p>Vertikalt eller horisontellt i systemet. En risk är till exempel att man bygger gränssnitt först och sedan kör man mellan-lager, sedan bygger man backend. Och det är en risk, då vet man inte - återigen Scrum säger att varje sak ska generera ett värde. Bygger jag bara frontend så genererar jag inget värde. Ja men kan jag logga in nu? Nej, men du har en login knapp och en ruta. Det genererar inget värde. Om vi stoppar projektet nu så har vi egentligen bara en login ruta visuellt, men vi har ingen funktion. Och då blir man slicad vertikalt istället. Så att - kunna logga, ja då gör vi kanske det i prompten istället. Nu kan du logga in, då har vi gjort det så att både användaren kan logga in, och det finns något implementerat inlogg i backend. Och när man leker med tankarna så skulle det kunna vara att - ja, första gång var det en textfil som kontrollerar vilken användaren är. Nästa gång så byter vi ut den mot en databas. Nästa gång ska databasen vara krypterad och så, man bygger på det hela tiden. Varje gång som man gör det här inkrementella steget så ska man generera ett nytt värde.</p>
43	I5	<p>Och det är väldigt svårt att följa, men jag tror också det är därför att - där har man på något sätt nog kommit närmre dem med cloudlösningar. Tidigare var det så att när man skulle implementera någonting - ja, men då måste vi först sätta upp miljön, och så ditten och datten i två veckor eller fyra veckor, som egentligen inte genererar något värde - efter två veckor så frågar kunden, och man har en miljö och en server som man kan deploya till. Okej, tack. Men med molntjänster så har man direkt - okej men vi deployar upp ett nytt - så ligger det uppe. Då helt plötsligt kan man jobba mer med att vara värde drivet på ett helt annat sätt. För det är också en risk som man har, och det är lite det som är en del av molntjänster, det är att de eliminerar riskerna på det sättet att man inte lägger för mycket tid på att preparera miljö innan du går ut på banan.</p>
44	D	<p>Och en annan grej som dyker upp i litteraturen är att det finns vissa artefakter som man använder för att hantera risker i agila metoder. Som till exempel en risk burndown chart, eller Kanban, risk register och liknande. Använder ni sådana verktyg för era projekt?</p>
45	I5	<p>Riskmatris använder jag inte. I alla fall inte jag, det är säkert någon här som kör det. Det handlar också lite om situationen man sitter i. Vissa projekt har inte så mycket risk i sig, då behöver man inte ta till dem verktygen. Och i vissa kanske märker man att här måste man belysa risken, man märker att det är det som tar stort fokus och det är det som kan hindra oss. I Scrum, alltså burndown chart kör vi hela tiden. Där är risken att - okej, vad är risken att vi inte klarar av att bli klar med det som vi ska committa inom sprinten? Risken för själva arbetet - okej nu har vi gått halva sprinten, vi har såhär mycket kvar och enligt denna prognos så kommer vi kanske inte kunna göra allting färdigt till reviewen. Ok, hur hanterar vi den risken? Ja men då kanske vi behöver se vilka user stories är det som är nästan färdiga, så vi kan se att vissa user stories är så pass lite gjord på så de kommer inte kunna bli färdiga till reviewen. Och då är det kanske dumt att lägga arbete på dem. Så tappar man några sådana</p>

		<p>som man inte skulle hinna med. Och i Kanban så har man risken att man vill belysa flaskhalsar. Det här med WIP (Work In Progress) limit för varje kolumn, varje del i processen ska man säga. Okej, här ser vi att oj, vi har en hög risk här. Vi får inte igenom lika många user stories som vi hade specificerat som vi kan implementera. En risk där, hur ska vi hantera det? För där ser man att flödet blir - vi får cycle time för varje user story har ett behov av att - vi har för lite resurser på testning till exempel. Har man den risken så kan man säga att - ja men då blir det flaskhals framför testning kolumnen, och saker som vi implementerade kommer inte vidare till testning, vi får trycka igenom det. Och då kan man säga att man hanterat det i stunden där. Det kör vi.</p>
46	D	<p>Och hur upplever du att er nuvarande riskhanteringsprocess är? Skulle man kunna behöva en lite mer explicit riskhanteringsprocess som traditionella metoder har? Eller räcker det att man jobbar agilt? Att man hanterat risker genom att jobba agilt?</p>
47	I5	<p>Jag tror mer på agilt. Jag tror inte att ha verktyg utanför hjälper det. Däremot ser jag mer att det som kan vara lite problem med agilt är att man - risken är att om man inte har kommunikation i ett Scrum team, så fallerar många saker. Man får inte samarbete, man får den här risken att user stories som skapas inte är detaljerade tillräckligt för att dem genererar ett värde som de skulle göra. Att man tappar acceptanskriterierna, att man tappar vad man ska göra, man gör för mycket eller för lite. Det är en stor risk där. Och det egentligen är risken att teamet inte kommunicerar. För ansvaret blir på teamet, och på att dem ska skapa en user story där alla är överens om vad bilden är och vad som ska åstadkommas med denna user story. Risken är där som är jättehög i Scrum och i agilt att man inte går tillräckligt djupt i den saken, att man börjar jobba med saker som man inte riktigt vet hur den är avgränsad. Tar man in user stories som inte är avgränsade så kan det betyda att - man börjar jobba på user stories som man tror är av mindre storlek, sedan helt plötsligt när man börjar jobba med den så växer den för att man inte har gått igenom och tittat på vad det är vi ska göra. Har vi avgränsat den? För det är en stor risk i Scrum och agilt, att man inte har avgränsat user stories. Att de är större än vad man kan hantera. Om man inte avgränsar dem så betyder det att man - det är för mycket saker i dessa user stories. De skulle egentligen brutits ner till mindre user stories. Men man har trott - ja men vi är här, det är ju solklart, och så kommer man ner till djupet av det och så märker man att det tar tid och kravet på den - det egentligen behöver brytas ner till flera user stories för att den är så pass stor. Det är en stor risk som finns i Scrum där. Och det är oftast att man går in och gör fel, det är att man har för stora user stories inne i sprintar. Det är också i Kanban, man har stora tasks i Kanban så man får inget kontinuerligt flöde, utan det - ena stunden är den liten och andra stunden är den stor. Det är det här att bryta ner och avgränsa. Avgränsning kan jag tycka är viktigast.</p>

48	D	När ni hanterar risker när ni jobbar agilt, anser du att det strider mot agila principerna? Agila manifestet values och principles.
49	I5	Nej, det tycker jag faktiskt inte. Det skulle jag inte påstå. Men risken där - jag tror det är mer att man belyser risker på olika sätt. Det som jag ser med value-drivet är att risken är att man inte genererar värde. Tidigare var det såhär att - risker för projekten är att man inte håller deadline eller sin budget. Och det har man sett i agilt att det är skit samma om de håller budgeten, det är egentligen bra ändå. Risken att du inte genererar det värdet som man har tänkt sig att det ska generera, och det är det hela agila går på, att du ska generera det värdet som man har tänkt sig att det ska generera.
X	X	Due to time constraint, the interviewee had to leave. The remaining questions were completed through email.
50	D	Anser du att en explicit riskhanteringsprocess (likt de som finns i traditionella vattenfallsprojekt; identification, assessment, planning, monitoring) är nödvändigt när man jobbar agilt? Varför?
51	I5	Ja. Det du nämner ovan finns med när man jobbar agilt. I Scrum jobbar man med att ha transparens i projektarbetet mellan inom team och mellan produkt owner. Detta genom att ha en prioriterad product backlog som alla kan se och som alla jobbar efter. Sprintplaning och grooming för framtida planering av kommande sprint. Arbetar med Daily Scrum för planering och skapa transparens inom projektet under pågående sprint. Och sprint review och retrospective för att utvärdera värdet som har genererats i en sprint i form av features. Velocity kan användas på olika sätt men ett vanligt sätt är att kunna göra prognos framåt hur mycket features Scrum teamet kan färdigställa under en sprint.
52	D	Påverkar riskhanteringsprocesser era agila arbetsmetoder? Blir ni mindre agila, eller är det stärkande att implementera riskhantering?
53	I5	Inom Agilt jobbar man tätare med riskhantering och har en större plats. Från design beslut till att effektivisera impl. Arbete.
54	D	Finns det projekt där ni inte använder er av riskhantering? Under vilka omständigheter skulle riskhantering inte behövas?
55	I5	Alla våra projekt innehåller riskhanteringsprocesser i mer eller mindre utsträckning
56	D	Till vilken utsträckning anser du att riskhantering har påverkat framgång eller misslyckanden av era projekt?
57	I5	Riskhantering har en stor betydelse. Jag tycker att inom Agilt jobbar man tätare och aktivt att eliminera risker som uppstår och återkoppling om risk är snabbare för berörda och hanteras löpande.
58	D	Om du har erfarenhet av traditionella metoder, hur skiljer sig riskhantering i agila projekt från traditionella projekt? Behöver agila metoder lika mycket riskhantering som traditionella projekt?
59	I5	Jag tror att riskhanteringsprocessen är lika stor men en stor skillnad är att traditionella metoder försöker förutspå de flesta riskerna medan agila belyser och hanterar verkliga risker som uppstår under projektarbetet.

8 References

- A guide to the project management body of knowledge : (PMBOK guide)*. (2016). Solna : WSP Sverige, 2016 5. utg.
- Albadarneh, A., Albadarneh, I., & Qusef, A. (2015). *Risk management in Agile software development: A comparative study*. Paper presented at the Applied Electrical Engineering and Computing Technologies (AEECT), 2015 IEEE Jordan Conference on.
- Alleman, G. B. (2002). *Agile project management methods for ERP: how to apply agile processes to complex COTS projects and live to tell about it*. Paper presented at the Conference on Extreme Programming and Agile Methods.
- Alvehus, J. (2013). *Skriva uppsats med kvalitativ metod : en handbok*: Stockholm : Liber, 2013 (Kina) 1. uppl.
- Andrat, H., & Jaswal, S. (2015). *An alternative approach for risk assessment in Scrum*. Paper presented at the Computing and Network Communications (CoCoNet), 2015 International Conference on.
- Avison, D. E., & Fitzgerald, G. (2006). *Information systems development : methodologies, techniques and tools* (4th ed.). London: McGraw-Hill.
- Bannerman, P. L. (2008). Risk and risk management in software projects: A reassessment. *Journal of systems and software*, 81(12), 2118-2133.
- Boehm, B. W. (1991). Software risk management: principles and practices. *IEEE Software*, 8(1), 32-41.
- Cervone, H. F. (2011). Understanding agile project management methods using Scrum. *OCLC Systems & Services: International digital library perspectives*, 27(1), 18-22.
- Chin, G. (2004). CHAPTER 8: APPROACHING RISK IN AN AGILE ENVIRONMENT. In (pp. 123-140): American Management Association International.
- Chow, T., & Cao, D.-B. (2008). A survey study of critical success factors in agile software projects. *Journal of systems and software*, 81(6), 961-971.
- Cohen, D., Lindvall, M., & Costa, P. (2004). An introduction to agile methods. *Advances in computers*, 62(03), 1-66.
- Coram, M., & Bohner, S. (2005). *The impact of agile methods on software project management*. Paper presented at the Engineering of Computer-Based

- Systems, 2005. ECBS'05. 12th IEEE International Conference and Workshops
- De Bakker, K., Boonstra, A., & Wortmann, H. (2010). Does risk management contribute to IT project success? A meta-analysis of empirical evidence. *International Journal of Project Management*, 28(5), 493-503.
- Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012, 06//). A decade of agile methodologies: Towards explaining agile software development, Editorial. *Journal of Systems & Software*, pp. 1213-1221.
- Elbanna, A., & Sarker, S. (2016). The Risks of Agile Software Development: Learning from Adopters. *IEEE Software*, 33(5), 72-79.
- Fowler, M., & Highsmith, J. (2001). *The agile manifesto*. *Software Development*, 9(8), 28-35.
- Gilb, T., & Finzi, S. (1988). *Principles of software engineering management*. Wokingham: Addison-Wesley.
- Glass, R. L. (2004). Matching methodology to problem domain. *Communications of the ACM*, 47(5), 19-21.
- Gold, B., & Vassell, C. (2015). *Using risk management to balance agile methods: A study of the Scrum process*. Paper presented at the Knowledge-Based Engineering and Innovation (KBEI), 2015 2nd International Conference on.
- Henninger, S., Ivaturi, A., Nuli, K., & Thirunavukkaras, A. (2002). *Supporting adaptable methodologies to meet evolving project needs*. Paper presented at the Conference on Extreme Programming and Agile Methods.
- Jacobsen, D. I., Sandin, G., & Hellström, C. (2002). *Vad, hur och varför : om metodval i företagsekonomi och andra samhällsvetenskapliga ämnen*: Lund : Studentlitteratur, 2002 (Lund : Studentlitteratur).
- Jiang, L., & Eberlein, A. (2009). *An analysis of the history of classical software development and agile development*. Paper presented at the Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on.
- Khatri, S. K., Bahri, K., & Johri, P. (2014). *Best practices for managing risk in adaptive agile process*. Paper presented at the Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions), 2014 3rd International Conference on.
- Kvale, S., Brinkmann, S., & Torhell, S.-E. (2009). *Den kvalitativa forskningsintervjun*: Lund : Studentlitteratur, 2009 (Ungern) 2. uppl.
- Lei, H., Ganjeizadeh, F., Jayachandran, P. K., & Ozcan, P. (2017). A statistical

- analysis of the effects of Scrum and Kanban on software development projects. *Robotics and Computer-Integrated Manufacturing*, 43, 59-67.
- Lindvall, M., Basili, V., Boehm, B., Costa, P., Dangle, K., Shull, F., . . . Zelkowitz, M. (2002). *Empirical findings in agile methods*. Paper presented at the Conference on extreme programming and agile methods.
- Maxwell, J. A. (2005). *Qualitative research design : an interactive approach*: Thousand Oaks, CA : Sage Publications, cop. 2005 -2. ed.
- McConnell, S. (1996). *Rapid Development: Taming Wild Software Schedules*: Microsoft Press.
- Moran, A. (2014). *Agile risk management*. New York: Springer.
- Moran, A. (2016). Overview Agile Risk Management Techniques.
<http://institute.agileriskmanagement.org/overview-agile-risk-management-techniques/>
[2017-06-05]
- Munassar, N. M. A., & Govardhan, A. (2010). A comparison between five models of software engineering. *IJCSI*, 5, 95-101.
- Nyfjord, J. (2008). *Towards integrating agile development and risk management*. Institutionen för data-och systemvetenskap (tills m KTH),
- Nyfjord, J., & Kajko-Mattsson, M. (2007). *Commonalities in risk management and agile process models*. Paper presented at the Software Engineering Advances, 2007. ICSEA 2007. International Conference.
- Patton, J. (2002). *Designing requirements: incorporating usage-centered design into an agile SW development process*. Paper presented at the Conference on Extreme Programming and Agile Methods.
- Project Management Institute (2009). *Practice Standard for Project Risk Management*. Project Management Institute, Inc.
- Project Management Institute (2015) *Capturing the Value of Project Management Through Organizational Agility*
- Sommerville, I. (2016). *Software engineering* (Tenth edition. ed.). Boston: Pearson.
- Walczak, W., & Kuchta, D. (2013). Risks characteristic to Agile project management methodologies and responses to them. *Operations Research and Decisions*, 23.
- Whitney, K.M. and Daniels, C.B., 2013. The root cause of failure in complex IT

projects: complexity itself. *Procedia Computer Science*, 20, pp.325-330.

Ylimannela, V. (2011). A model for risk management in agile software development. *Tampere University of Technology*.