DEGREE PROJECT IN THE FIELD OF TECHNOLOGY
INFORMATION AND COMMUNICATION TECHNOLOGY
AND THE MAIN FIELD OF STUDY
COMPUTER SCIENCE AND ENGINEERING,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2018

The Agile method Scrum in development of safety critical applications

A case study about challenges and opportunities for developers and verifiers

KIM HILTUNEN

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

The Agile method Scrum in development of safety critical applications – A case study about challenges and opportunities for developers and verifiers

KIM HILTUNEN

Master in Computer Science
Date: December 9, 2018
Supervisor: Hamid Reza Faragardi
Examiner: Olof Bälter
Swedish title: Den Agila metoden Scrum vid utveckling av säkerhetskritiska applikationer - En fallstudie om utmaningar och möjligheter för utvecklare och verifierare
School of Electrical Engineering and Computer Science

Abstract

When it comes to using agile methods in safety critical application development, there is a limited amount of empirical findings. To learn more about how people in this field perceives the use of working with these methods, it is of interest to take part of their experiences and opinions.

The purpose of this thesis is to discover advantages, disadvantages and improvement factors of working with the agile method Scrum in combination with safety critical application development. The study was limited to the roles of developers and verifiers working in two anonymized companies in the defence and railway industry. A qualitative approach was used which included a multiple case study where each of the involved company were considered a case. Empirical data was collected through semi structured interviews with the employees from the two companies. The collected data was categorized, coded and analyzed using comparative analysis. The data was coded based on one of the seven areas documentation, organization, communication, education, development, verification and planning.

The interviewed developers and verifiers pointed out various advantages, disadvantages and improvement factors within the areas mentioned above. The majority of the opinions among the interviewees varied. However, some common aspects were pointed out. The most frequently mentioned factor to improve for the developers was in the communication area, while the verifiers raised educational aspects as the most common factor to improve.

The findings from this study can be used to point out sections that the investigated companies should consider when using the agile method Scrum in combination with safety critical application development. The thesis also provides empirical evidence of how people in the inspected companies consider difficulties and opportunities in their work.

Keywords: Agile methods, Scrum, Safety critical application development

Sammanfattning

När det kommer till användadet av agila metoder vid säkerhetskritisk applikationsutveckling finns det begränsat med empiriska fynd. För att få veta mer om hur personer som arbetar inom detta område uppfattar användandet av dessa metoder, är det av intresse att ta del av deras erfarenheter och åsikter.

Syftet med denna uppsats är att identifiera fördelar, nackdelar och förbättringsmöjligheter när det kommer till att arbeta agilt med Scrum vid säkerhetskritisk applikationsutveckling. Studien inkluderade rollerna utvecklare och verifierare som arbetade i två anonymiserade företag inom försvars- respektive järnvägsindustrin. En kvalitativ metod användes vilket inkluderade en fallstudie, där de involverade företagen behandlades som varsitt fall. Empirisk data samlades in genom semistrukturerade intervjuer med anställda från de två företagen. All insamlad data kategoriserades, kodades och analyserade med hjälp av komparativ analys. Kodningen utfördes baserat på de sju olika områdena dokumentation, organisation, kommunikation, utbildning, utveckling, verifikation och planering.

De intervjuade utvecklarna och verifierarena pekade ut diverse fördelar, nackdelar och förbättringsfaktorer inom områdena som nämndes ovan. Majoriteten av åsikterna varierade, däremot kunde några gemensamma åsikter påvisas. Den mest förekommande förbättringsfaktorn bland utvecklarna var inom kommunikationsområdet, medan verifierarna tog upp utbildningsrelaterade aspekter som den vanligaste förbättringsfaktorn.

Resultaten från denna studie kan användas för att peka ut delar som de undersökta företagen bör beakta vid användning av den agila metoden Scrum i kombination med säkerhetskritisk applikationsutveckling. Uppsatsen tillhandahåller empiriska bevis på hur personer som arbetar på de granskade företagen ser på svårigheter och möjligheter inom sitt arbete.

Nyckelord: Agila metoder, Scrum, Säkerhetskritisk applikationsutveckling.

Acknowledgement

I would like to thank my two supervisors Hamid Reza Faragardi (KTH) and Andreas Kristensson (Combitech) for their support throughout the thesis writing.

Thank you!

Contents

List of Figures

List of Tables

Acronyms

ALARP As Low As Reasonably Possible.

ASD Adaptive Software Development.

DSDM Dynamic Systems Development Methods.

EN European Standard.

FDD Feature-driven Development.

GAMAB Global Au Moins Aussi Bon.

IEC International Electrotechnical Commission.

MEM Minimum Endogenous Mortality.

QA Quality Assurance.

SIL Safety Integrity Level.

STPA System Theoretic Process Analysis.

THR Tolerable Hazard Rate.

XP Extreme Programming.

Chapter 1

Introduction

1.1 Background

An application is defined as safety critical if its failures has the risk to harm humans, environment or lead to financial loss [1] [2]. Safety critical applications can for instance be found in airplanes and trains.

When it comes to working with development of safety critical applications there are standards that are followed, for example International Electrotechnical Commission (IEC) 61508 and European Standard (EN) 50128 [3]. There are different standards depending on in which industry the software development is taking place. An example of this is EN 50128 that is used in the railway industry, because it is specified for the development of software used in railway control systems [4]. Currently, a lot of these standards are adapted to fit older development methods such as the waterfall model [5]. The waterfall model consists of several phases including management of requirements, software design, implementation and software testing. All of these phases are methodically completed in order [6]. However, the standards does not require the sequential execution of the model to be strictly interpreted, which enables the use of other development methods such as agile ones. Agile development methods are described as a collection of system development methods based on the Agile Manifestos principles and values [7]. Traditional development models, which includes the waterfall model, are not as adaptable to changes related to customer needs and requirements in the projects as agile development methods [6] [5]. For instance, a risk of using tradi-

tional development models could be if issues linked to customer needs or project requirements are found at a later phase of the project. In this case there is a risk that the project has to go back to an earlier project phase, which means that parts, or in some cases the whole project, has to be remade. This is not the case in projects where agile methods are used, since these projects are able to adapt to changes at any time. The use of agile methods in safety critical application development could for example lead to a lowered development cost and allow projects to handle changes in an more efficient way compared to traditional development models [5] [8]. Agile methods allows the projects to involve multiple roles simultaneously and have work executed in parallel. These methods in combination with safety critical application development has a lot of challenges that must be considered and adapted, for example being able to handle parts related to safety [8]. By taking part of experience, opinions and knowledge among the people who work agile with safety critical application development, it is possible to identify additional challenges they are facing and which improvement factors that could be done to facilitate their work.

The developers task is to write and manage code that is used in applications. A verifiers work is based on proving that requirements stated in the standards have been fulfilled for the implemented work. The agile method Scrum, which is investigated in this thesis, is considered a framework for development of various advanced products such as software [9].

1.2 Objective

The objective of the thesis is to identify advantages, disadvantages and improvement factors for developers and verifiers who are working with the agile method Scrum in combination with safety critical application development. The study will investigate two companies from the defence and railway industry.

1.3 Research Question

The thesis aims to investigate and answer the following research questions:

- *Which advantages and disadvantages are developers and verifiers facing during development of safety critical applications where the agile method Scrum is used?*
- *What is needed to facilitate the use of the agile method Scrum in a role as a verifier or developer?*

1.4 Delimitations

The thesis will focus on verifiers and developers who work with safety critical application development from two different companies. These two roles were of interest for the companies, which is why they became central for this thesis. The study also has its focus on software development which involves the agile method Scrum, because it is the method used in the investigated industries. The standards used for development of safety critical applications will mainly focus on the standards IEC 61508 and EN 50128 in this study.

1.5 Research Methodology

The research methodology for this thesis is stated in Figure 1.1. The first task consisted of defining the research questions and goals of the thesis. In order to do so, previous studies were identified and reviewed. A literature review of previous work was performed within the fields of agile methods, development of safety critical applications and the standards EN 50128 and IEC 61508. A detailed presentation of the remaining methods used in the thesis are explained in chapter 4.

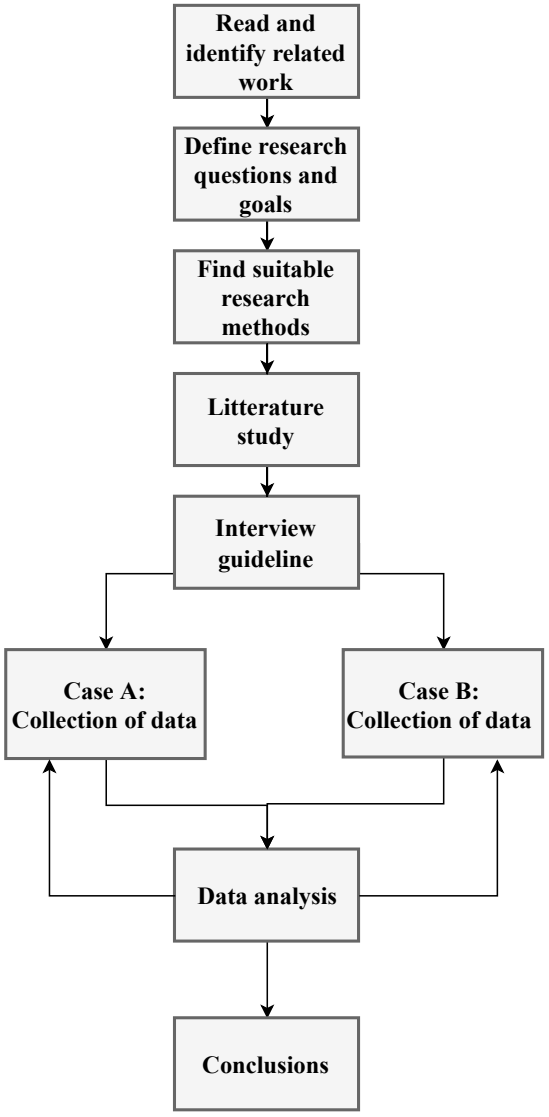

Figure 1.1: Research methodology

1.6 Contribution

The contributions of this study is identification of drawbacks and promoting aspects of using Scrum in development of safety critical applications for the roles of verifiers and developers. There is a limited amount of empirical studies within this field that highlights how developers and verifiers working with safety critical application development perceives the use of Scrum. This work contributes to pointing

out the opinions and viewpoints of how it is to work with Scrum as a developer or verifier in development of safety critical applications. Overall, the findings could improve the development and adaptation of Scrum in such projects. It is also of interest to have empirical findings on how Scrum suits for this kind of development in the investigated companies, which the thesis provides.

1.7 Outline

Chapter 1 introduces the background, thesis objective, research method, research questions and delimitations of the work.

Chapter 2 presents background theory within the field of agile methods, Scrum and safety critical application development.

Chapter 3 presents previous work.

Chapter 4 displays the methods used for the thesis which includes a qualitative method, multiple case study and semi-structured Interviews. The analysis method that consists of coding, categorization and a comparative analysis is also presented. The chapter ends with an ethical discussion.

Chapter 5 shows results from the collected data.

Chapter 6 presents the analysis and discussion of results. The chapter also features a review of the validity.

Chapter 7 includes a conclusion of the thesis and a proposal of future work.

Chapter 2

Theoretical Background

2.1 Agile Methods

As stated in the introduction, agile development is described as a collection of system development methods that are based on the Agile Manifestos principles and values [7]. Examples of agile methods are Extreme programming (XP), Scrum, Feature-driven development (FDD), Dynamic Systems Development Methods (DSDM), Crystal and Adaptive Software Development (ASD) [6][1]. The Agile Manifesto was created and signed in 2001, by a group of seventeen people who at the time worked with software development [10]. The group, known as the Agile Alliance, felt that they needed an option for the software development processes at that time, which resulted in the Agile approach. Agile development is not solely fit to handle software development and can be used for various types of product development [11]. The Agile Manifesto presents four core values and twelve principles that should be followed when working agile, with any of the methods included.

2.1.1 Agile Values

The Agile Manifesto presents four values that are used when working agile. The values can be seen in List 2.1.

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

List 2.1: Values from the Agile Manifesto [10]

Highsmith [10] writes: "That is, while there is value in the items on the right, we value the items on the left more".

2.1.2 Agile Principles

The Agile Manifesto introduces twelve principles that should be followed when working agile. The principles are displayed in List 2.2.

- "Our highest priority is to satisfy the customer through early and continuous delivery of valuable software."
- "Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage."
- "Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale."
- "Business people and developers must work together daily throughout the project."
- "Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done."
- "The most efficient and effective method of conveying information to and within a development team is face-to-face conversation."
- "Working software is the primary measure of progress."

- "Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely."
- "Continuous attention to technical excellence and good design enhances agility."
- "Simplicity–the art of maximizing the amount of work not done–is essential."
- "The best architectures, requirements, and designs emerge from self-organizing teams."
- "At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly."

List 2.2: Principles from the Agile Manifesto [12]

2.2 Scrum

The agile method Scrum was created by Ken Schwaber and Jeff Sutherland in the 1990s [9]. The method can be seen as a framework used for development of products. Scrum is performed in an iterative and incremental way, and is described as lightweight, easy to interpret and hard to master. The two creators of Scrum have together written the Scrum guide, that presents the definition and cores of Scrum including rules, roles, events and artifacts. It is possible to include other processes in combination with Scrum [9]. Scrum is based on empiricism, which means that knowledge is obtained from experience. The gained knowledge is used as a base for decision making. Empiricism relies on the three keystones: transparency, inspection and adaptation. Transparency strives for full visibility of crucial parts of the work for the people involved in it. Inspections of processes included in the work are carried out continuously to avoid any unwanted changes. Adaptations of the work is done if any inspection shows that there are parts that should not be included in the working process [9] [13].

2.2.1 Roles in Scrum

A Scrum Team includes the roles of a Product Owner, a Development Team and a Scrum Master. A Scrum Team has two attributes. The first one is that it is self-organizing, which means that it is the teams decision how the work is performed. The second attribute that the Scrum Teams have is cross-functionality. A cross-functional team includes enough expertise, making it possible to execute all necessary work within the team. As a result of this, the Scrum Team will not need any help from people not included in the team [9].

The Product Owner is a person who represents the stakeholders for the product that is developed [13]. The Product Owner is responsible for handling the Product Backlog and making sure that content of it is prioritized [9]. An explanation of the Product Backlog is presented in subsection 2.2.3.

The Development Team consists of people that performs the work of the increments for the product. Based on recommendations, the size of the Development Team is between three and nine people. A larger or smaller team size can cause problems with the work [9]

The Scrum Masters task is to make sure that all activities of the work performed by the Scrum Team is carried out according to the existing rules,principles and values of Scrum. The Scrum Master can be seen as a support role to the Scrum Team by serving each role in various ways to improve their work [9].

2.2.2 Events in Scrum

In Scrum the work is divided into Sprints. A Sprint is a time limited section of the work where an increment of the final product is produced. The time of a Sprint may vary, but the maximum time is set to a month. When a Sprint is finished, the next one will start as soon as the Sprint Review and Sprint Retrospective are completed. The Product Owner has the possibility to cancel a sprint if the work done in it is not needed for the final product [9].

A Sprint starts with a Sprint Planning. In the initial Sprint Planning meeting, the Scrum Team decides what the Sprint should result in and

what work that the group has to do in order to reach the goals of the Sprint [9].

The Daily Scrum is a meeting where the Development Team is able to plan their work for the upcoming working day. The Development Team has to look back on what they have done since the last Daily Scrum meeting and adjust their plan according to it.

Before a Sprint ends a Sprint Review is conducted. The objective of the Sprint Review is to look over and demonstrate the increment created in the latest Sprint and if needed apply changes to the Product Backlog. The Sprint Review also includes a general inspection of Sprint related parts such as the project timeline, budget, future work and encountered problems during the sprint and how they were solved. After the Sprint Review an updated version of the Product Backlog is generated, that includes all changes stated during the review. The Scrum Master is responsible for the occurrence of the Sprint Review and has to make sure that the time-limit of it is kept [9].

The final event before a Sprint ends is a Sprint Retrospective. It can be seen as a self evaluation of the Scrum Team and their work carried out in the latest sprint. If the Scrum Team discovers that any changes has to be done related to the way they work, the team is able to apply the changes for the upcoming Sprints. The Scrum Team can also use parts of the work that went well for the forthcoming Sprints. The Sprint Retrospective is an event made to improve the future work that the Scrum Team performs [9]. An overview of the Scrum process is displayed in Figure 2.3

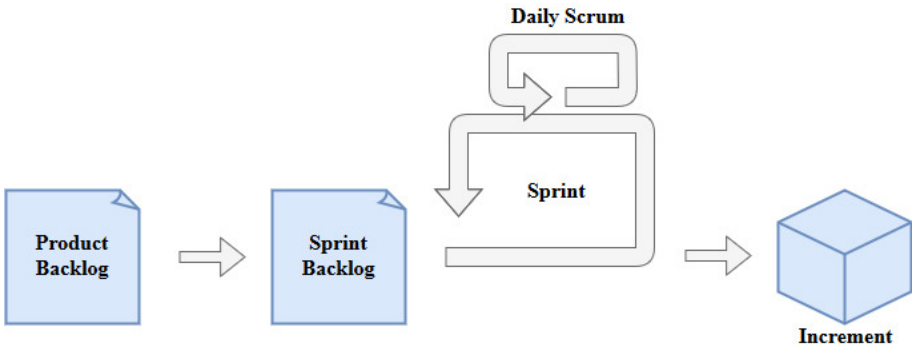

Figure 2.3: Overview of the Scrum process [9]

2.2.3 Artifacts in Scrum

The Product Backlog is a list where the product requirements are stated [13][9]. As mentioned earlier, it is the Product Owners responsibility to handle the Product Backlog. The Product Backlog can and most likely will be changed during the working process, which mean that it is possible to add or remove requirements to make them fit the work being performed [9].

A Sprint Backlog can be compared with the Product Backlog, but instead of covering the whole work of a product the Sprint Backlog only includes items from the Product Backlog that are part of a Sprint. The Sprint Backlog also states what the Scrum Team has to do to achieve their goals of the Sprint and develop an increment of the product. The Sprint Backlog is created and maintained by the Development Team [9].

2.3 Safety Critical Applications

When a failure of a system has the chance to harm humans or cause any kind of environmental damage, the system is considered safety critical [1] [2]. A failure is defined as when a system does not execute a function or service in a planned way that is expected by the user [14]. There are two different types of failures, systematic failures and random failures. Systematic failures are used to describe a broad range of failures such as manufacturing defects, specification mistakes, implementations errors and issues related to operation maintenance [15]. Random failures are for example caused by pressure on a hardware device [15]. There is a possibility for a random failure to happen at any time during the lifetime of a hardware device.

2.3.1 Safety Integrity Level (SIL)

Safety Integrity Level (SIL) is defined as one of four levels based on intervals of failure probabilities of a safety function [16]. SIL is used in the standards IEC 61508 and EN 50128. There are different scales for the SIL level in each standard, where IEC 61508 has a scale from SIL 1 to SIL 4 and EN 0128 has a scale from SIL 0 to SIL 4 [17] [4]. The SIL 0 level in EN50128 is added to ensure that a minimum level of safety

integrity always is fulfilled, since there are uncertain aspects during risk evaluation and hazard identification [4].

To determine the SIL level, the Tolerable Hazard Rate (THR) must be calculated. The dangerous events that are caused by equipment are measured by the THR [18]. There are different methods that can be used to calculate the THR such as Global Au Moins Aussi Bon (GAMAB), As low as reasonably possible (ALARP) and Minimum endogenous mortality (MEM) [19]. After a THR value is calculated, it is possible compare the value to fixed tables in the standards. For example IEC 61508 has two different tables for deciding the SIL level which depends on if the safety-related system in operated in low, high- or continuous mode. High demand mode means that the demand that requires a safety function to be used is set to at least once per year or more. Low demand mode means that there is a demand of the safety function being used that is not greater than once per year. The continues mode is described as a state that demands the safety function to be used continuously [16] [20].

2.4 Standards

IEC defines a standard in the following way:

> *"A standard is a document, established by consensus and approved by a recognized body, that provides, for common and repeated use, rules, guidelines or characteristics for activities or their results, aimed at the achievement of the optimum degree of order in a given context."* [21]

Laws, regulations and directives are often written without any technical details regarding how to develop or use software in safety critical systems [22]. The European Union indites European directives and each member state is expected to form its laws and regulations accordingly. The technical details that handles development and the use of software in safety critical systems is described in the standards. There are harmonized standards issued by a standardization organization such as CENELEC. If a standard is harmonized it fulfils a certain directive. Each standardization organization writes standards for var-

ious industries. It is not mandatory to follow standards while developing a product and manufacturers may show in their own way how parts related to the safety of a product meets all directives, regulations and laws [22].

2.4.1 IEC 61508

The standard IEC 61508 presents a general method for all activities included in the safety life cycle of systems, that consists of parts that are electrical and/or electronic and/or includes programmable electronic [17]. The standard is divided into seven parts displayed in Table 2.1, where each part handles specific topics.

Part of IEC 61508	**Topic**
1	General requirements
2	Requirements for electrical/electronic/programmable electronic safety-related systems
3	Software requirements
4	Definitions and abbreviations
5	Examples of methods for the determination of safety integrity levels
6	Guidelines on the application of IEC 61508-2 and IEC 61508-3
7	Overview of techniques and measures

Table 2.1: Parts included in IEC 61508 [22]

2.4.2 EN 50128

The standard EN 50128 is used in the railway industry for control and safety applications and defines the process and requirements needed for programmable electronic systems [4]. The standard is only specified for processes included in safety critical software development in the railway industry.

Chapter 3

Previous Work

This chapter presents previous work done within the field of agile methods used in development of safety critical software.

3.1 An Exploratory Study on Applying a Scrum Development Process for Safety-Critical Systems

The paper presents a customized version of Scrum named S-Scrum that includes a new technique used for hazard analysis, System Theoretic Process Analysis (STPA) [23]. In the study the authors are conducting a case study. that shows empirical evidence on how Scrum combined with STPA can be used for development of safety critical applications.

3.2 Challenges and Opportunities in Agile Development in Safety Critical Systems – A Survey

In this survey the challenges and opportunities of agile development methods in combination with safety critical systems development is investigated [24]. The subjects consists of people from the areas of safety and agile development and 31 practitioners within fields such as railway, health and automotive answered a questionnaire. The study

presents that 80% of the respondents agreed that agile methods can be used in combination with safety critical system development. Other results that the authors highlights relates to importance of compliance with standards, support for an iterative assessment of the safety case process during the development of software, guarantee of product safety and the management of safety requirements.

3.3 Application of an Agile Development Process for EN50128/railway conformant Software

This paper proposes a customized version of Scrum called Safe Scrum, that is adapted to fit development of safety critical systems [8]. The aim of the paper is to examine how well Safe Scrum fits for development of EN 50128 compliant software. The sections 5 to 7 of EN 50128 are analyzed and critical parts when it comes to working agile are pointed out. The sections that the authors points out as critical are presented in Table 3.1. Their conclusion of the paper is that the use of Safe Scrum enables projects to be conformant with EN 50128, if the critical parts are handled. A solution on how to handle each of the issues found in each section of the standard is proposed in the paper. This study has a theoretical approach and none of the findings are tested empirically. The paper points out parts that could be critical when using agile methods for safety critical applications, which could show possible parts of interest that could be pointed out by the interviewees in this thesis.

Section number	Subject
5.1.2	Organization
5.3.2	Lifecycle issues
6.1.4	Test requirements
6.2.4	Software verification requirements
6.5	Software quality assurance
6.6	Modification of change control
7.1	Life cycle and documentation for generic software
7.2	Software requirements
7.4	Component design
7.5	Component implementation and testing

Table 3.1: Critical sections of EN 50128 when using Safe Scrum [8]

3.4 Scaling Agile Methods to Regulated Environments: An Industry Case Study

The objective of the study is to investigate how the use of agile methods can meet the standards used in regulated environments [39]. The investigated company, QUAMAS, works with quality and compliance management for the life science industry. The authors defines regulated environments as domains where standards and/or regulations and/or other directives must be satisfied. In the paper, key concepts of environments that are regulated are compared with the use of agile methods and a discussion about possible issues that might occur when working with the approach is presented. The methods used for data collection in the study includes semi-structured interviews with key roles in the company, read documentation related to the process of software development and access to the tools used during the software development process. The authors inclusion of several types of data collection, referred to as triangulation, is done to increases the validity of the study. An interesting conclusion that Fitzgerald et al. presents is that developers and roles that involves quality assurance (QA) in general has a positive attitude of using agile methods and its benefits. In this case the use of agile methods in safety critical software development was very successful and adapted in way that suited the company involved in the study.

3.5 The application of Safe Scrum to IEC 61508 certifiable software

Each section of the thirds part of IEC 61508 and the requirements in them are evaluated, to control how well Safe Scrum fits software development based on the standard [35]. The work is performed in two iterations where the authors mark each section with "OK", "?" or "Not OK". The "OK" mark is used when the section does not require any adjustment to Scrum or the standard. The mark "?" is used if the section needs further discussion among the authors to be properly evaluated. "Not OK" is used for the sections that requires any kind of changes to fit Scrum. By isolating all non-Scrum related activities, or as the authors defines as "separation of concern" in the text, and remove them from the evaluation in the second iteration, the authors are able to narrow down the number of issues in need of adjustment. The final assessment of issues can be seen in Table 3.2.

Section number	**Subject**	**Number of requirements in section marked as critical**
7.1	How to structure the development of the software	2
7.3	How to develop a plan for validating the software safety	2
7.4.2	How to create, review, select, design and ensure the safety of the system	9
7.4.7	Requirements for software module testing	1
7.9	How to test and evaluate the outputs from a software safety lifecycle	1

Table 3.2: Critical sections of IEC 61508 when applying Safe Scrum [8].

3.6 Agile vs. plan-driven in safety-critical development cases (a clash of principles?)

In a multiple case study involving two companies in the medical industry, an investigation of possible clashes between the core values of plan-driven and agile methods is examined [25]. The thesis aims to answer the research question: "How is the discussion regarding agile vs. plan-driven development in safety-critical projects reflected in practice? ". The authors are collecting data from interviews made with people from the companies and by analysis of documents including project plans. Coding is used to analyse the data which generates tables that presents the results. Examples of some of the clashes the authors finds relates to documentation, changing requirements, communication and planning. The authors draws the conclusion that there are issues when using agile methods for safety critical development. They also point out that the biggest obstacle to overcome to be able to combine agile methods and safety critical development is to manage the risks in a proper way.

Chapter 4

Method

4.1 Qualitative Method

In this study a qualitative research approach was chosen. A qualitative approach is suitable for investigating behaviour of humans, causes of why certain things occur and the thoughts, experiences and knowledge among people [26]. In this work the thoughts and experiences of working with the agile method Scrum in safety critical application development for the two roles of developers and verifiers were being investigated, which was the reason to adopt a qualitative approach.

4.2 Multiple Case Study

If the research questions are defined in an exploratory way, a case study is considered a suitable strategy according to Yin [27]. The research questions of this work aimed to explore a specific topic, which made the use of a case study relevant. Since two different companies were investigated a multiple-case study was conducted, where each company were seen as a case. The company within the railway industry is referred to as Case A or and the company in the defence industry is considered as Case B. The two companies were selected to participate in this study because they both work agile with safety critical application development.

4.2.1 Anonymity

To reduce any form of negative consequences caused by the information presented by the companies or the subjects included in the study, a choice was made to perform an anonymization. This included a de-identification of the companies and participants names to reduce the possibility to connect certain information to a specific company or individual. In the thesis the companies were referred to as Company A and Company B based on the same pattern that was presented in section 4.2.

4.3 Semi-structured Interviews

Interviewing is common technique used to collect empirical data in qualitative studies [28]. Yin points out that interviews are one of the most important ways to collect empirical data in case studies, which therefore was done in this study [27]. The interviews were semi-structured, which enabled the use of open-ended questions to get to know more about the thoughts of the subjects [29]. The interviewees were developers and verifiers working with development of safety critical applications from the two companies presented in section 4.2.

An interview invitation was written in both Swedish and English and sent by email to people working as developers or verifiers in two companies. The interview invitation contained information about the topic, reason and objective of the study, a brief presentation of myself, data confidentiality, interview settings and conditions and contact information for further questions related to the thesis. The interview invitation can be seen in Appendix A.

An interview guide was created to have topics and questions for discussion. It is presented in Appendix B. The interview guide had an introduction to get to know more about the participant and his or her role and previous experience of agile work. After the introduction the main questions were put and divided into different areas. These areas were documentation, organization, planning, verification, development, communication, knowledge and competence, general and other. Each area consisted of questions related to the specific section. The areas for the interview guide were based on findings from two

previous studies by Myklebust et al. and Jacobsen et al. and also what could be thought to occur in the work for each of the roles that were investigated in this thesis [25] [8].

The interviews were carried out face-to-face or on Skype. All the interviews were recorded to enable a proper analysis of the data. The recordings for the face-to-face interviews were done on two devices. By recording on two devices, the risk of losing data was reduced.

All interviews were transcribed and the participants were offered to read the transcription from their interview. This is a technique used in qualitative research and is known as member check or member validation [30] [31] [32]. The technique is used to increase the validity of the data and to make sure that the interview is interpreted correctly. Member check can be used in different steps of the research process [30]. In this thesis it has only been used after the interviews. Not all interviewees wanted to control their interview transcription.

4.4 Analysis Method

Coding, categorization and comparative analysis were used to analyze the gathered empirical data from the interviews. The three concepts are further explained in the subsections below.

4.4.1 Coding

As a first step of analyzing the interview data, coding was used. This method involves the assignment of so called codes to arrange data [33] [29]. The interview transcripts were coded based on the areas included in the interview guide. The areas consisted of documentation, organizational, communication, education, development, verification and planning. For example when a part of the interview that the subject talked about was related to documentation, the code for that specific section was set to documentation.

4.4.2 Categorization

Categorization is an analysis method used to cluster data findings such as codes into categories or blocks [33]. The data from the in-

terviews were put in one of the categories advantages, disadvantages and improvement factors. The three categories were taken from the research questions of this study.

4.4.3 Comparative analysis

Comparative analysis is used to find equality or diversity by comparison between parts such as interviews, groups, individuals or cases [33]. The analysis method is commonly used in case study research, to perform comparison between different cases. In this study two comparisons were carried out. The first one analyzed each of the roles of developer and verifier isolated. The second comparison was done between the two cases, Case A and Case B.

4.5 Ethics

All interviewees participated voluntarily in this study. They all gave consent to attend and in the interview invitation, the attendees were informed about interview conditions such as the purpose of the study, how data would be handled, professional secrecy and anonymization. These factors are also mentioned by Trost in his chapter about ethics [34].

The collected data from the interviews included personal opinions from the participants. To avoid any issues related to the interviewees personal opinions, the interview transcripts were de-identified. This involved the removal possible connections in transcripts that could uncover an individuals identity or reveal the names of the companies that were involved in the study.

4.6 Sustainability

Sustainability can be divided into three sections: social, ecological and economical sustainability. From a social point of view, by considering the ideas presented by the participants in this study, an increased effectivity of working with Scrum in safety critical application development could be achieved. The work environment could gain from this, which would benefit the employees well-being. An economical aspect

that is linked to an increased effectivity, is that companies could benefit economically due to decreased expenses. The ecological aspects are not affected by this work.

Chapter 5

Empirical Data

This chapter presents the empirical data from the interviews with participants from the companies included in the study. Table 5.1 presents all the participants that has been interviewed in this study. In total four developers and four verifiers were interviewed from both companies.

There were participants who had roles such as Product Owner and Scrum Master in the Scrum Teams, while other subjects were members of the Development Team.

Company	Role
A	Developer
A	Developer
A	Developer
A	Verifier
A	Verifier
B	Developer
B	Verifier
B	Verifier

Table 5.1: Participants from interviews in the thesis

5.1 Case A

This section presents the data collected for the company in the railway industry. The Scrum work experience ranged from 3 months to 2

years among the respondents in Case A. All the participants worked primarily with the standard EN 50128.

5.1.1 Case A - Advantages

The three developers from Case A mentioned that there are promoting factors related to **documentation**. The documentation is performed in parallel with all other tasks included in a sprint. In this way the documentation becomes a natural part of the work and is performed iteratively, which leads to a constant progress of the documentation. One developer said

> *"...now it has been easier to really acquire that for each development step, the documentation is sort of a natural part of it all"*.

One developer stated an **organizational** improvement that an agile process has lead to. There is now a better connection and collaboration between testers and developers. This has resulted in that the two roles now are in phase with each other when it comes to their work. Earlier, a developer that added changes to the code would have to wait until a tester at a later stage of the project approved or disapproved the code. This caused the developer not to know if any further changes would be necessary until the tester responded. One of the verifiers was on the same track and stated that the work now involves a closer collaboration with other roles and groups. In this way it is easier to receive feedback at an earlier stage of the work.

A verifier thought that an agile way to work has resulted in the **verification** always being up to date, due to a better overview of the current state of the project.

When it came to **communication** there were several benefits mentioned by the interviewees. One developer said that the direct communication between people in the project has increased as an outcome of the daily meetings. Another developer stated that there now is an increased communication between various roles such as developers, system engineers and testers. A third developer thought that Scrum has lead to a clearer definition of the person responsible for informing the Scrum Team in comparison to the way they worked before. This enables the Scrum Team to receive relevant information which facilitates

their work. A verifier told that communication is clearer compared to the way they worked before, due to the fact that information now reaches the Scrum Team directly during the daily meetings.

When the topic of **education** was discussed during an interview one developer said that their Scrum Team were given a Scrum coach that was present for a couple of weeks. This was a advantage since the team members were able to start discussions with the coach when needed.

In the **development** area, a developer stated that an agile approach has resulted in a faster work progress, higher effectivity and is cheaper. Another developer said that the overall quality of the work has increased. The same developer also told that each individual in the Scrum Team now takes more responsibility of their work due to personal assignment of tasks among the Scrum Teams. This has ended up in enhanced code quality.

5.1.2 Case A - Disadvantages

Among the developers there were disadvantages related to **documentation**. One developer stated that there are many people that types in the same document which leads to difficulties to read the document. Another developer thought that it is possible to perceive an agile approach as slower compared to the previous way they worked, because it had changed the way they worked with the documentation. The respondent mentioned that the documents related to their tasks always should be in releasable condition. This requires that another person verifies any changes performed. This could lead to a feeling of that there more steps to go through to complete a task, since they before could have done several changes and verified them all on the same time.

Drawbacks linked to **communication** were noticed by several of the interviewees. One verifier said that it is difficult to keep in contact with different teams, since it requires the subject to establish contact with numerous Product Owners and Scrum Masters. One developer and verifier told that there is a lack of necessary information to team members in some of the Scrum Teams. The verifier also said that the information from specific parts of the organization to the verifier is

limited, which hinders the subjects work. One more issue caused by lack of communication between the Scrum Teams presented by a developer was that it could lead to that some tasks are forgotten and not handled by any team and that tasks are worked on by multiple teams, which results in that two teams does the same job twice. Another developer said that it was due to unclear apportions of the tasks between the Scrum Team that lead to that some tasks were missed and not worked on. Another disadvantage linked to communication was mentioned by one of the verifiers. There is a risk that information is filtered by Scrum Masters and Product Owners, since they are the ones that receives the information from the project management first. This could lead to that the team members does not acquire relevant information. One of the developers pointed out that there could be a possible cultural barrier between the team members, since some of the people from other countries would not ask for help in the same extent as the members from the same country as the subject.

An **organizational** disadvantage that a verifier mentioned is the overhead caused by the use of cross functional teams. This way to structure the work groups was used in the company earlier and the subject pointed out that this lead to that each team had their own leader. Another section related to organization that the same verifier pointed out was a possible management issue. The subject is not sure if management has looked in EN 50128 an adapted the Scrum Teams based on the standards description of each role. The participant questioned whether all roles in the standards were represented in the Scrum Teams. One developer told that high expectations of effectiveness on an agile way of working from the executives at the company could lead to unrealistic time plans. Another respondent said that projects sometimes are started without all necessary parts in place, for example the requirements.

Two developers said that there are disadvantages to have team members spread out at different locations and offices. The first developer told that this lead to put focus on parts that not are related to problem solving such as documentation and the use of tools. The other developer said that the same issue caused a loss of communication to the team members not present at the office.

One developer thought that an agile approach has lead to that the workload shifts for different roles during the sprints. The participants gave an example when the team knows that they have two weeks it could lead to that certain roles tend to adjust their work flow to this time schedule, even though the work could have been finished in one week. The subject did not point out any specific role and said that in one sprint one role could have less work and in another sprint the some role could have more work compared to the other roles.

One **planning** related disadvantaged presented by one of the developers was that it is hard to break down and finish all the planned tasks in a sprint because the sprints are too short. The subject also said that there are many phases to complete before finishing tasks.

A developer said that it is hard to synchronize the work, because different tasks requires a varied amount of time. This was considered an issue related to **development**. The participant gave an example that the time it took to implement a requirement could range from one day to two weeks. This could lead to that they waited for input from someone else while they were finished with a task.

An **educational** drawback that a developer mentioned were that there has been a lack of Scrum education. The subject stated that they instead used on the job training to get knowledge about Scrum.

5.1.3 Case A - Improvement factors

The following parts were proposed by the interviewees in Case A in order to facilitate their work:

- Two developers would prefer to have all team members located at the same office. One developer wanted all team members present at the workplace at least two times per week to facilitate the communication in the team.
- One developer requested a tool to keep track of what has been implemented on specific lines of the code.
- Two developers and one verifier wanted more Scrum related education.

- One developer wanted to know more how senior roles fits into an agile way of working.
- Introduction of new tools to maximize the usage of it was requested by one of the developers.
- A coach to support the discussions in the team was suggested by one developer.
- A developer preferred better conditions for the team meetings because some of the tools that currently are used for communication does not work that well.
- One respondent would like to have a weekly meeting with other teams to discuss possible changes in the work.
- One developer thought that it was needed to increase the awareness of positive aspects of working as a team among the team members.
- Introduce a project leader in each Scrum Team that can manage tasks that has been forgotten or that are not handled by any other Scrum Team.
- A developer stated that the sprints were too short, which made it problematic to finish the required work. Analyzing the regression tests and document handling were time consuming and two week sprints was not enough to finish enough of the work.
- A verifier thought that smaller Scrum Teams are better compared to bigger ones. The subject also suggested that it is better to split bigger teams into smaller ones to simplify the work.
- One verifier also requested to have a finished project plans before starting a project. Sometimes the projects are the started before everything is in place that is needed to execute the work.
- One respondent also wanted to be more flexible and use other methods depending on what part of a project they worked on. The subject said that there could be parts where an agile way of working would not fit that well.

- A verifier would prefer a single contact person responsible for communication to people not included in any of the Scrum Teams. This would result in that no one not part of any Scrum Team would not have to contact all Scrum Masters and Product Owners to obtain information. The contact person would act as an link between the Scrum Teams and the person not part of any team and forward all relevant information to its recipient.
- One developer said that when they first started to work with Scrum, there were unclear instructions of the different groups responsibilities. A start-up meeting would ease the transition from the old way to work to Scrum.
- One verifier would like to have an increased responsibility among the Scrum Teams when it comes to documentation. There have been issues where the code has been implemented but the documentation has not been finished.
- One developer wanted to have more information and to have all the required input available before any decision making is done. The respondent said that the lack of information could lead to that the team started to work with parts that are less prioritized or performed the work incorrect. The subject suggested a role who would facilitate the dissemination of information.

List 5.1: Case A: Improvements presented by interviewees

5.2 Case B

This section presents the data collected for the company in the defence industry. The Scrum work experience ranged from 1 year and 3 months to 3 years among the respondents in Case B. The standards used in Case B varied among the participants. The standards that were used in Case B were IEC 61508, MIL-STD-498 and standards created by the company in case B.

5.2.1 Case B - Advantages

One verifier said that an agile way of working promoted an iterative progress with the **documentation**. This made it possible to avoid afterwards documentation of work done up to a year ago. A developer said that they now have more documentation compared to the way they worked before. This was a positive aspect, because it made it simple to follow the development of a product and see why certain decisions were taken along the way.

An **organizational** advantage that a verifier mentioned was that cross functional teams educated each other. The respondent gave an example of how developers were involved in verification and learned how to verify and how verifiers were included in software development and learned that part of work. The developer told that there is a good team spirit in the Scrum Team. The team members are able to cover up for each other if needed and they share their issues and progress in the daily meetings. Another verifier also stated that the daily meetings made it easier to follow the progress of other roles and groups.

One verifier and one developer thought that an agile way of working promoted the overall **communication**. The verifier said that it increases the effectivity of the projects, because of the possibility to at an earlier stage manage specific sections of work. The verifier also stated that the daily meetings were a crucial part of boosting the effectiveness, as it made it possible to react on parts of the work that had an negative impact on the project and steer it in the right direction.

The developer thought that one advantage is that there are shorter loops and faster feedback during the software **development**. The interviewee said:

> *You work a little bit more in cycles rather than in a solid waterfall where every department hands over a finished document of how it should look like. We are working more in cycles and are more reactive to what is happening.*

The developer also told that there now is an opportunity to influence parts of the work around oneself, because everything is not carved in stone and not finished at every point of delivery.

5.2.2 Case B - Disadvantages

The increased amount of **documentation** has lead to a drawback reported by the developer. It takes a lot of time to perform the documentation and analyze the documents. Another document related disadvantage presented by one of the verifier was that an iterative way of documenting only adds small changes to the main documents. This restricts the overall view of the documents.

The developer pointed out that there is a risk of failure if there is a lack of **communication** in the project. An agile approach requires open communication between various groups involved in the work.

A **planning** based drawback that one of the verifier expressed was that it is hard to move around the deadline for certain tasks while working agile. At their department it is required that at a certain date, the requested functionality must be implemented. The developer told that there are long loops when issues are found in the construction specification, which makes it difficult to get any changes applied.

When software **development** was discussed in an interview one respondent told that all parts are not always well thought-out and defined, which lead to issues with the software development.

An **organizational** disadvantage stated by a verifier was how the Scrum Retrospectives that were held after the Sprint were to repetitive and that the same questions and issues appeared every retrospective meeting. The subject said that the questions and issues raised in the Scrum Retrospectives were related to parts of work the Scrum Team always could improve. Due to this the team decided to remove the retrospective meetings. Another organizational drawback presented by one of the verifiers was that the Scrum Teams work in their own bubble, which lead to problems with the coordination between different teams. A final disadvantage connected to organization that was pointed out in Case B was also presented by one of the verifiers. The participant told that security levels in the company obstructed the use of certain tools in projects.

Shorter loops or Sprints compared to before makes it difficult for ver-

ifiers to finish their manual **verification**, according to one of the verifiers. The developers produces more code than the verifiers are able to verify before new code is sent to them.

5.2.3 Case B - Improvement factors

List 5.2 displays the improvements that were proposed by the interviewees in Case B.

- One verifier wished for more automation of verification to avoid a bottleneck in the verification.
- A verifier thought that an introduction course in Scrum could improve the knowledge of the employees.
- One verifier was interested of "Lessons learned" from other projects, since it would be valuable to know more about experience from other projects.
- The developer wanted to improve and simplify the communication between all groups in the project, because at the moment there are groups that could share essential information to other groups.
- The developer also thought that an increased effectiveness in the communication between different groups, such as system verification, to shorten the feedback loops could lead to improved results.

List 5.2: Case B: Improvements presented by interviewees

Chapter 6

Analysis and Discussion

As noted in Chapter 5 there were a lot of advantages, disadvantages and improvement factors presented by the interviewees, which was the objective of this thesis. The subsections 5.1.1, 5.1.2, 5.2.1 and 5.2.2 displays the answers to RQ1, while subsections 5.1.3 and 5.2.3 shows the answer to RQ2.

The comparative analysis showed that all the developers mentioned at least one advantage related to documentation, development and communication. Three out of four developers stated how an iterative work with the documentation had affected their work positively. In the development area the answers differed among the developers. One common topic that two of the developers mentioned was the positive effects of getting feedback during the development phase. When it came to communication all the developers stated that an agile approach had increased and improved the overall communication in various ways, for example more communication between different roles in the project and within the Scrum Teams. There were no common disadvantages mentioned by all of the developers. However, some of the developers pointed out issues related to the areas organization, documentation and communication. An organizational drawback that was pointed out by two of the developers was that there were team members spread out at different locations, not working from the same office. Communication was the only area that was presented as an improvement factor by every developer. Examples that were lifted by the interviewees are better conditions for team meetings and the importance to share relevant information to various teams.

There were no general areas suggested by all of the verifiers when it came to the categories advantages and improvement factors. The most frequent section for the advantages was connected to organization, which was presented by three out of four verifiers. A topic that two of them told about was the positive aspects of the cross-functional teams. An area that every verifier presented as an disadvantage was related to organization. However, none of them raised any common issues. Individual standpoints from the participants pointed out possible management issues when creating the Scrum teams, difficulties with coordination between the Scrum teams, company security levels causing problems to introduce new tools and that projects were started without having essential parts such as requirements in place. The improvement factors presented by the verifiers showed that the most usual area was connected to education. Two out of four verifiers requested more Scrum education and one interviewee wanted to know more about experience from other projects, a so called "lessons learned".

The participants in both of the cases reported the organizational aspect as the most usual disadvantage. It could be of interest for the companies to look over these issues in order to make possible improvements in the organizational area. The participants from Case A proposed eleven possible improvements, while the interviewees from Case B did not suggest any improvement factors at all. This could have to do with the fact that some adjustments already have been done within this area in Case B. Another element that was noted in the data from Case A, was that communication was a considered an issue. Participants in Case A stated communicational issues such as cultural barriers between team members and lack of information flow between and within the Scrum Teams. This could have to do with Case A having team members located abroad, which also is raised as an organizational disadvantage by two participants in Case A. Having team members located at different locations contributes to the absence of direct contact between team members. However, interviewees from Case B does not perceive the same communicational or organizational issues. A possible success factor in Case B could be the usage of smaller teams with members located at the same office and the opportunity to have a direct, physical communication. A solution for Case A to avoid com-

municational issues could be to set up teams were the members are located in the same office, in order to simplify the direct contact between team members. A potential drawback of rearranging the teams in this way could be that some teams loses experience from members located abroad. There were indications of that some of the employees does not receive necessary information. In order to improve communication, it is essential to establish channels in the companies where all the employees and teams can receive information that is important to accomplish their work.

There are some aspects related to the research method that should be discussed. Due to non continuous contact with different contact persons for each company, there were difficulties to obtain interviews. There were issues concerning security, weather it was possible to conduct the interviews and which employees from the companies that would participate in the study. This caused a delay of the collection of empirical data and an uneven balance of participants in both of the cases. This lead to that more data was presented for Case A compared to Case B and that it was impossible to find data patterns for developers in Case B.

Since the interview questions were not sent out to the subjects before the interviews, there is a possibility that the participants did not have enough time to reflect on their answers. This could have affected the collected data from the interviews, as crucial information can have been missed.

The comparative analysis was difficult to apply with a limited amount of data, which was the case in this study. The possibility to find patterns in the data was restricted. Due to the limited amount of data, the results of this study cannot be generalized outside of the investigated context.

As seen in the in the previous work chapter 3 there is a limited amount of empirical finding of using Scrum in combination with safety critical application development. Many of the previous studies mainly includes theoretical findings.

None of the critical sections from the standards EN 50128 and IEC

61508 pointed out in the studies by Myklebust et al. and Stålhane et al. were mentioned by any of the interviewees of this thesis [8] [35]. There were a limited amount of answers from the respondents that pointed out any issues related to any standard. However, as stated earlier one verifier in Case A pointed out that there could be a possible issue with the Scrum Team setup that management assembled in their company. The interviewee wondered whether management had looked into the role description of EN 50128 when the Scrum Teams were put together.

One of the findings that Doss et al. presented was that 80% of the participants in their study agreed on that agile methods can be used in combination with safety critical system development [24]. The interviewees in this thesis raised critical parts of working agile, even though none of them said that they were against this way to work. However, all of the respondents suggested factors that can be improved from various parts of their work.

In the thesis written by Jacobsen et al., the empirical findings from two companies in the health care industry indicated that there were issues of using agile methods in combination with safety critical development cases [25]. Some of the areas that were pointed in their study were also noted in this thesis. These areas concerned documentation, planning and communication.

One of the agile principles displayed in List 2.2 states the following: "Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done" [12]. The employees in this study points out various issues and improvement proposals within different areas. Considering their issues and proposals could promote employees motivation and further develop the working method.

Another agile principle that was presented in the theoretical background stated: "The most efficient and effective method of conveying information to and within a development team is face-to-face conversation" [12]. As stated earlier, there are participants who raises issues with not having the possibility to meet their team members in person. Therefore, this is something that the Case A company could review further.

6.1 Validity of Results

Validity aims to show that the indented purpose of the study really is measured [34]. The questions that were used in the interviews strived for getting information on which advantages, disadvantages and improvement factors developers and verifiers face during safety critical application development using an agile approach. All the interviewees work within the field of interest, even though their experience of working agile with safety critical application development varied.

The data from the interviews is not open to the public due to anonymization and the chance to reveal sensitive information of the companies. A disadvantage of anonymization that the Swedish Research Council points out is the difficult to verify empirical data in the study [36]. This could affect the validity of the study, since it is not possible to review the interview transcripts. There is a risk that the information in the interview transcripts have been misinterpreted when analyzed, which could have influenced the outcome of the thesis. To minimize this risk member checking was used after the interviews in order to increase the validity of the collected data and to make sure that the transcriptions were correct. However, it is important to note that not all participants wanted to check their interview transcript.

Trost states that the use of questions that include statements and the lack of usage of follow up questions can decrease the reliability of a study [34]. Therefore, follow up questions have been used and no questions including statements have been asked in this thesis.

Chapter 7

Conclusion

The findings from this study highlighted a lot of interesting aspects for developers and verifiers that could be considered in development projects of safety critical applications where Scrum is used.

The developers in this study pointed out that there were advantages when it came to working with Scrum in safety critical application development. The advantages were connected to documentation, communication and development. However, some of them also mentioned disadvantages within documentation and communication. There were also drawbacks presented when it came to organization. The only common area for the improvement factors that all the developers mentioned was associated with communication.

The most frequently mentioned area of advantages for the verifiers was linked to organization. A common organizational topic that was raised by two of the verifiers was the positive effects with having cross-functional teams. The verifiers presented several organizational disadvantages, even though none of them raised any common issues. The educational aspect was pointed out as an improvement factor among certain verifiers, where two of them requested more Scrum related education.

The comparison between the two cases, Case A and Case B, showed that disadvantages related to organizational aspects were the most mentioned one among the participants. The result also showed that there were some differences between the cases. The interviewees from

Case A mentioned to a greater extent problems lined to communication, while the participants from Case B did not raise this as a common issue. One possible cause to this could be that Case B had all team members located at the same office and that they in some cases had smaller teams.

The presented results from this study could point out sections that companies in the railway or defence industry could consider when the agile method Scrum is used in safety critical application development. It is however important to note that the use of Scrum in development of safety critical applications is a continuous developmental process. One step in this process could be to collect information about co-workers observations and ideas in order to be able to solve occurring problems and take advantage of improvement ideas. This thesis shows an example of how the involved participants are motivated to contribute to such a process.

7.1 Future Work

In this thesis the focus has been on the two roles of verifiers and developers in safety critical application development projects. To get a better picture of how different roles finds the use of agile methods in similar projects, it would be of interest to study other roles as well. Another area of interest for a future study could be to investigate how other agile methods such as XP are perceived among employees in industries that work with safety critical application development.

Bibliography

[1] G. Xiaocheng, R. F. Paige, and J. A. McDermid. "An iterative approach for development of safety-critical software and safety arguments". In: *AGILE Conference, 2010*. IEEE, 2010, pp. 35–43.

[2] M. Rausand. *Reliability of Safety-Critical Systems: Theory and Applications*. en. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2014. ISBN: 978-1-118-77635-3 978-1-118-11272-4. DOI: 10.1002/9781118776353. URL: http://doi.wiley.com/10.1002/9781118776353 (visited on 08/06/2018).

[3] B. Douglass. "Agile Development for Embedded Systems". en. In: *Software Engineering for Embedded Systems*. Elsevier, 2013, pp. 731–766. ISBN: 978-0-12-415917-4. DOI: 10.1016/B978-0-12-415917-4.00021-9.

[4] European Committee for electrotechnical standardization (CENELC). *Railway applications - Communications, signalling and processing systems - Software for railway control and protection systems*. Standard 50128. 2011.

[5] T. Stålhane, T. Myklebust, and G. K. Hanssen. "Safety standards and Scrum – A synopsis of three standards". en. In: (2013), p. 13. URL: https://www.sintef.no/globalassets/safety-standards-and-scrum_may2013.pdf (visited on 02/23/2018).

[6] M. Holcombe. *Running an Agile Software Development Project*. en. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2008. ISBN: 978-0-470-38588-3 978-0-470-13669-0. DOI: 10.1002/9780470385883. URL: http://doi.wiley.com/10.1002/9780470385883 (visited on 03/28/2018).

[7] Agile Alliance. *What is Agile Software Development?* en-US. 2015. URL: https://www.agilealliance.org/agile101/ (visited on 03/20/2018).

[8] T. Myklebust, T. Stålhane, and N. Lyngby. "Application of an Agile Development Process for EN50128/railway con-formant Software". In: (2015). DOI: 10.1201/b19094-529. URL: https://www.researchgate.net/profile/Thor_Myklebust/publication/281587160_Application_of_an_Agile_Development_Process_for_EN50128railway_conformant_Software/links/55eed78d08ae199d47bf6876/Application-of-an-Agile-Development-Process-for-EN50128-railway-conformant-Software.pdf.

[9] K. Schwaber and J. Sutherland. *The Scrum Guide*. 2013. URL: http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf (visited on 03/04/2018).

[10] J. Highsmith. *Manifesto for Agile Software Development*. 2001. URL: http://agilemanifesto.org/ (visited on 03/06/2018).

[11] P. Measey. "Agile Foundations - Principles, Practices and Frameworks". In: (2015). ISSN: 978-1-78017-254-5. URL: https://app.knovel.com/hotlink/toc/id:kpAFPPF001/agile-foundations-principles/agile-foundations-principles.

[12] J. Highsmith. *Principles behind the Agile Manifesto*. 2001. URL: http://agilemanifesto.org/principles.html (visited on 03/20/2018).

[13] K. Schwaber. *Agile project management with Scrum*. Redmond, Wash. : Microsoft Press, 2004. ISBN: 0-7356-1993-X.

[14] M. Kraeling. "Safety-Critical Software Development". en. In: *Software Engineering for Embedded Systems*. Elsevier, 2013, pp. 613–645. ISBN: 978-0-12-415917-4. DOI: 10.1016/B978-0-12-415917-4.00018-9. URL: http://linkinghub.elsevier.com/retrieve/pii/B9780124159174000189 (visited on 08/06/2018).

[15] M. Gentile and A. E. Summers. "Random, systematic, and common cause failure: How do you manage them?" en. In: *Process Safety Progress* 25.4 (2006), pp. 331–338. ISSN: 1547-5913. DOI: 10.1002/prs.10145. URL: http://onlinelibrary.wiley.com/doi/abs/10.1002/prs.10145 (visited on 08/06/2018).

[16] I. E. C. (IEC). *IEC 61508: Functional Safety - FAQ ed2.0*. 2018. URL: http://www.iec.ch/functionalsafety/faq-ed2/page5.htm?iecfaq=8 (visited on 04/11/2018).

[17] International Electrotechnical Commission (IEC). *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 3: Software requirements*. Standard 61508-3. 2010.

[18] D. H. Schäbe. "Different Principles Used for Determination of Tolerable Hazard Rates". en. In: (2001), p. 8.

[19] P. Wigger. "Experience with Safety Integrity Level (SIL) Allocation in Railway Applications". en. In: (2001), p. 16.

[20] International Electrotechnical Commission (IEC). *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 5: Examples of methods for the determination of safety integrity levels*. Standard 61508-5. 2010.

[21] I. E. C. (IEC). *IEC - Standards development > International Standards (IS)*. 2018. URL: http://www.iec.ch/standardsdev/publications/is.htm (visited on 02/19/2018).

[22] Försvarets materielverk (FMV). *H ProgSäk 2018 (Handbok för programvara i säkerhetskritiska tillämpningar)*. 2018. URL: https://www.fmv.se/Global/Dokument/Verksamhet/Systems%C3%A4kerhet/Kurser/H_ProgSak_2018_Fastst%C3%A4lld.pdf (visited on 02/01/2018).

[23] Y. Wang, J. Ramadani, and S. Wagner. "An Exploratory Study on Applying a Scrum Development Process for Safety-Critical Systems". In: *Product-Focused Software Process Improvement*. Ed. by M. Felderer et al. Vol. 10611. Cham: Springer International Publishing, 2017, pp. 324–340. ISBN: 978-3-319-69925-7 978-3-319-69926-4. DOI: 10.1007/978-3-319-69926-4_23. URL: http://link.springer.com/10.1007/978-3-319-69926-4_23 (visited on 03/03/2018).

[24] O. Doss and T. P. Kelly. "Challenges and Opportunities in Agile Development in Safety Critical Systems: A Survey". en. In: *ACM SIGSOFT Software Engineering Notes* 41.2 (May 2016), pp. 30–31. ISSN: 01635948. DOI: 10.1145/2894784.2894798. URL: http://dl.acm.org/citation.cfm?doid=2894784.2894798 (visited on 03/03/2018).

[25] B. Jacobsen and M. Norrgren. "Agile vs. Plan-driven in safety-critical development cases-A clash of principles". In: (2008).

[26] K. Kelly and B. Bowe. "Qualitative Research in Engineering Education". en. In: Vancouver, Canada, 2011, p. 10.

[27] R. K. Yin. *Case study research : design and methods*. eng. 5. ed.. London: SAGE, 2014. ISBN: 978-1-4522-4256-9.

[28] C. B. Seaman. "Qualitative methods in empirical studies of software engineering". In: *IEEE Transactions on Software Engineering* 25.4 (1999), pp. 557–572. ISSN: 0098-5589. DOI: 10.1109/32.799955.

[29] K. E. Newcomer, H. P. Hatry, and J. S. Wholey, eds. *Handbook of practical program evaluation*. en. Fourth edition. Essential texts for nonprofit and public leadership and management. San Francisco: Jossey-Bass & Pfeiffer Imprints, Wiley, 2015. ISBN: 978-1-118-89360-9.

[30] L. Birt et al. "Member Checking: A Tool to Enhance Trustworthiness or Merely a Nod to Validation?" en. In: *Qualitative Health Research* 26.13 (2016), pp. 1802–1811. ISSN: 1049-7323, 1552-7557. DOI: 10.1177/1049732316654870. URL: http://journals.sagepub.com/doi/10.1177/1049732316654870 (visited on 07/31/2018).

[31] L. E. Koelsch. "Reconceptualizing the Member Check Interview". en. In: *International Journal of Qualitative Methods* 12.1 (2013), pp. 168–179. ISSN: 1609-4069, 1609-4069. DOI: 10.1177/160940691301200105. URL: http://journals.sagepub.com/doi/10.1177/160940691301200105 (visited on 07/31/2018).

[32] G. Ahrne and P. Svensson. *Handbok i kvalitativa metoder*. sv. 2:1. Författarna och Liber AB, 2015. ISBN: 978-91-47-11224-1.

[33] L. Given. *The Sage encyclopedia of qualitative research methods*. Los Angeles: SAGE, 2008. ISBN: 978-1-4129-4163-1.

[34] J. Trost. *Kvalitativa intervjuer*. sv. 4:1. Studentlitteratur AB, 2010. ISBN: 978-91-44-06216-7.

[35] T. Stålhane, T. Myklebust, and G. K. Hanssen. *The application of Safe Scrum to IEC 61508 certifiable software*. Tech. rep. 2012, p. 10. URL: https://pdfs.semanticscholar.org/55d9/6495405f44ddff830fb6589pdf.

[36] Swedish Research Council. *Good Research Practice*. 2017. ISBN: 978-91-7307-354-7.

[37] SINTEF / NTNU. *About SafeScrum*. en-US. 2015. URL: http://safescrum.no/sample-page/ (visited on 02/25/2018).

[38] W. C. Adams. "Conducting Semi-Structured Interviews". en. In: *Handbook of Practical Program Evaluation*. Ed. by K. E. Newcomer, H. P. Hatry, and J. S. Wholey. John Wiley & Sons, Inc., 2015, pp. 492–505. ISBN: 978-1-119-17138-6. DOI: 10.1002/9781119171386.ch19. URL: http://onlinelibrary.wiley.com.focus.lib.kth.se/doi/10.1002/9781119171386.ch19/summary (visited on 02/25/2018).

[39] B. Fitzgerald et al. "Scaling agile methods to regulated environments: An industry case study". In: *2013 35th International Conference on Software Engineering (ICSE)*. 2013, pp. 863–872. DOI: 10.1109/ICSE.2013.6606635.

[40] J. Highsmith. *History: The Agile Manifesto*. 2001. URL: http://agilemanifesto.org/history.html (visited on 03/28/2018).

[41] W. G. Gulland. "Methods of Determining Safety Integrity Level (SIL) Requirements - Pros and Cons". en. In: *Practical Elements of Safety*. Ed. by F. Redmill and T. Anderson. London: Springer London, 2004, pp. 105–122. ISBN: 978-1-85233-800-8 978-0-85729-408-1. DOI: 10.1007/978-0-85729-408-1_6. URL: http://link.springer.com/10.1007/978-0-85729-408-1_6 (visited on 06/13/2018).

[42] D. J. Smith and K. G. Simpson. "The Meaning and Context of Safety Integrity Targets". en. In: *Safety Critical Systems Handbook*. Elsevier, 2011, pp. 3–20. ISBN: 978-0-08-096781-3. DOI: 10.1016/B978-0-08-096781-3.10001-X. URL: http://linkinghub.elsevier.com/retrieve/pii/B978008096781310001X (visited on 08/06/2018).

Appendix A

Interview Invitations

A.1 English Interview Invitation

A.1.1 Case A

INTERVIEW INVITATION

Request to participate in a study regarding the use of the agile method Scrum when developing safety critical applications

As a step to make development of safety critical applications more effective, the use of agile methods is of interest. Standards such as IEC 61508 and EN 50128 do not require the use of the traditional waterfall model, which enables the use of an agile approach. By using agile methods, it is possible to involve multiple roles simultaneously and execute work in parallel. The use of such methods has a lot of challenges that must be considered and adapted when it comes to development of safety critical applications.

The purpose of the study is to investigate the use of the agile methods Scrum in development of safety critical applications for developers and verifiers. This includes identifying which obstacles and potentials developers and verifiers are facing during development of safety critical applications with an agile approach involving Scrum.

My name is Kim Hiltunen and I am currently studying the masters' program in computer science at KTH. This study is a part of my masters' thesis and is conducted in co-operation with Combitech.

The thesis includes interviewing individuals which are occupied within the field mentioned above. The interviewees must fulfil the following requirements to participate in the study:

- Work or worked as a developer or a verifier.
- Work or worked with the standards IEC 61508 and/or EN 50128.
- Work or worked with the agile method Scrum in safety critical application development.

The interviews will take approximately 45-60 minutes to conduct and will take place at your office. We will jointly agree upon a date in April or which is suitable for an interview. The participance is optional and you have the possibility to end the interview whenever you want. All the collected data will be handled with confidentiality and your name will not be mentioned in the report. To make it easier to perform data analysis, the interviews will be recorded and transcribed. The recordings will be deleted when the thesis is completed. The data from the interview will be presented in my master's thesis at KTH. A copy of the report will be sent to you when it is finished.

By participating in the study, you will contribute to highlight pros and cons of working agile with Scrum in safety critical development. This could lead to more effective planning of such development projects.

If you want more information about the thesis or if you have any questions, please contact me.

Kim Hiltunen
kimhil@kth.se

Supervisor:
Andreas Kristensson
andreas.kristensson@combitech.se

A.1.2 Case B

INTERVIEW INVITATION

Request to participate in a study regarding the use of the agile method Scrum when developing safety critical applications

As a step to make development of safety critical applications more effective, the use of agile methods is of interest. Standards such as IEC 61508 and EN 50128 do not require the use of the traditional waterfall model, which enables the use of an agile approach. By using agile methods, it is possible to involve multiple roles simultaneously and execute work in parallel. The use of such methods has a lot of challenges that must be considered and adapted when it comes to development of safety critical applications.

The purpose of the study is to investigate the use of the agile methods Scrum in development of safety critical applications for developers and verifiers. This includes identifying which obstacles and potentials developers and verifiers are facing during development of safety critical applications with an agile approach involving Scrum.

My name is Kim Hiltunen and I am currently studying the masters' program in computer science at KTH. This study is a part of my masters' thesis and is conducted in co-operation with Combitech.

The thesis includes interviewing individuals which are occupied within the field mentioned above. The interviewees must fulfil the following requirements to participate in the study:

- Work or worked as a developer or a verifier.
- Work or worked with the standards IEC 61508 and/or EN 50128 and/or any military standard such as MIL-STD 498.
- Work or worked with the agile method Scrum in safety critical application development.

The interviews will take approximately 30-45 minutes to conduct and will take place on Skype. We will jointly agree upon a date in May which is suitable for an interview. The participance is optional and you have the possibility to end the interview whenever you want. All the collected data will be handled with confidentiality and your name will not be mentioned in the report. To make it easier to perform data analysis, the interviews will be recorded and transcribed. The recordings will be deleted when the thesis is completed. The data from the interview will be presented in my master's thesis at KTH. A copy of the report will be sent to you when it is finished.

By participating in the study, you will contribute to highlight pros and cons of working agile with Scrum in safety critical development. This could lead to more effective planning of such development projects.

If you want more information about the thesis or if you have any questions, please contact me.

Kim Hiltunen
kimhil@kth.se

Supervisor:
Andreas Kristensson
andreas.kristensson@combitech.se

A.2 Swedish Interview Invitation

A.2.1 Case A

INTERVJUFÖRFRÅGAN

Förfrågan om att delta i en studie om användandet av den agila metoden Scrum vid utveckling av säkerhetskritiska applikationer.

I ett steg att göra utvecklingen av säkerhetskritiska applikationer mer effektiv är användandet av agila metoder av intresse. Standarder som IEC 61508 och EN 50128 ställer inga krav på att den traditionella vattenfallsmodellen används, vilket möjliggör användandet av ett agilt arbetssätt. Genom att använda agila metoder är det möjligt att involvera olika roller samtidigt och utföra arbete parallellt i projekt. Användandet av dessa metoder har en del utmaningar som behöver tas i beaktan och anpassas när det kommer till utveckling av säkerhetskritiska applikationer.

Syftet med denna studie är att undersöka användandet av den agila metoden Scrum för utvecklare och verifierare som deltar vid utveckling av säkerhetskritiska applikationer. Detta omfattar att identifiera vilka svårigheter och möjligheter utvecklare och verifierare ställs inför vid säkerhetskritisk applikationsutveckling där den agila metoden Scrum används.

Jag heter Kim Hiltunen och studerar för tillfället datalogi på master-nivå på KTH. Denna studie är en del av mitt examensarbete och utförs i samarbete med Combitech.

Arbetet inom området som nämns ovan inkluderar intervjuer. Intervjudeltagare måste uppfylla följande krav för att delta i studien:

- Arbetar eller har arbetat som utvecklare eller verifierare.
- Arbetar eller har arbetat med standarderna IEC 61508 och/eller EN 50128.
- Arbetar eller har arbetat med den agila metoden Scrum vid utveckling av säkerhetskritiska applikationer.

Intervjuerna beräknas ta 45–60 minuter att utföra och kommer att äga rum på din arbetsplats. Vi kommer tillsammans överens om ett passande datum i april eller maj då intervjun utförs. Deltagandet är frivilligt och du kan när som helst avbryta ditt deltagande utan närmare motivering. All data som samlas in kommer att hanteras konfidentiellt och ditt namn kommer inte att nämnas i rapporten. För att göra det enklare att genomföra dataanalysen, kommer intervjuerna att spelas in och transkriberas. Inspelningarna kommer att raderas när uppsatsen är slutförd. Data från intervjuerna kommer att presenteras i mitt examensarbete på KTH. När rapporten är färdig skickas en kopia till dig.

Genom att delta i studien bidrar du till att belysa fördelar och nackdelar av att arbeta agilt med Scrum vid säkerhetskritisk applikationsutveckling. Detta kan leda till en mer effektiv planering av denna typen av projekt.

Om du vill ha mer information om uppsatsen eller har några frågor, vänligen kontakta mig.

Kim Hiltunen
kimhil@kth.se

Handledare:
Andreas Kristensson
andreas.kristensson@combitech.se

A.2.2 Case B

INTERVJUFÖRFRÅGAN

Förfrågan om att delta i en studie om användandet av den agila metoden Scrum vid utveckling av säkerhetskritiska applikationer.

I ett steg att göra utvecklingen av säkerhetskritiska applikationer mer effektiv är användandet av agila metoder av intresse. Standarder som IEC 61508 och EN 50128 ställer inga krav på att den traditionella vattenfallsmodellen används, vilket möjliggör användandet av ett agilt arbetssätt. Genom att använda agila metoder är det möjligt att involvera olika roller samtidigt och utföra arbete parallellt i projekt. Användandet av dessa metoder har en del utmaningar som behöver tas i beaktan och anpassas när det kommer till utveckling av säkerhetskritiska applikationer.

Syftet med denna studie är att undersöka användandet av den agila metoden Scrum för utvecklare och verifierare som deltar vid utveckling av säkerhetskritiska applikationer. Detta omfattar att identifiera vilka svårigheter och möjligheter utvecklare och verifierare ställs inför vid säkerhetskritisk applikationsutveckling där den agila metoden Scrum används.

Jag heter Kim Hiltunen och studerar för tillfället datalogi på master-nivå på KTH. Denna studie är en del av mitt examensarbete och utförs i samarbete med Combitech.

Arbetet inom området som nämns ovan inkluderar intervjuer. Intervjudeltagare måste uppfylla följande krav för att delta i studien:

- Arbetar eller har arbetat som utvecklare eller verifierare.
- Arbetar eller har arbetat med standarderna IEC 61508 och/eller EN 50128 och/eller någon militär standard (exempelvis MIL-STD 498).
- Arbetar eller har arbetat med den agila metoden Scrum vid utveckling av säkerhetskritiska applikationer.

Intervjuerna beräknas ta 30–45 minuter att utföra och kommer att äga rum på Skype.
Vi kommer tillsammans överens om ett passande datum i maj då intervjun utförs. Deltagandet är frivilligt och du kan när som helst avbryta ditt deltagande utan närmare motivering. All data som samlas in kommer att hanteras konfidentiellt och ditt namn kommer inte att nämnas i rapporten. För att göra det enklare att genomföra dataanalysen, kommer intervjuerna att spelas in och transkriberas. Inspelningarna kommer att raderas när uppsatsen är slutförd. Data från intervjuerna kommer att presenteras i mitt examensarbete på KTH. När rapporten är färdig skickas en kopia till dig.

Genom att delta i studien bidrar du till att belysa fördelar och nackdelar av att arbeta agilt med Scrum vid säkerhetskritisk applikationsutveckling. Detta kan leda till en mer effektiv planering av denna typen av projekt.

Om du vill ha mer information om uppsatsen eller har några frågor, vänligen kontakta mig

Kim Hiltunen
kimhil@kth.se

Handledare:
Andreas Kristensson
andreas.kristensson@combitech.se

Appendix B

Interview Guide

B.1 English Interview Guide

INTERVIEW GUIDE

Introduction

[Start recording]

- Can you give a brief description of your working role?
  a) How does a regular working day look like?
  b) What is your function in a project?
  c) For how long have you been working in your current role?

- What agile methods have you been working with during safety critical application development?
  - For how long have you been working agile in safety critical application development?

- Are you working with IEC61508 / EN50128 or any other standard?

Main questions
DOCUMENTATION
1. What kind of documentation do you carry out in a project?
a) In what way does an agile way of working affect your way of documenting?

ORGANIZATION
1. Is there anything organization that can be done in order to facilitate an agile way of working? (meetings, group organization etc.).

PLANNING
1. How is the planning done in a project?
a) What parts are planned in your part of the work and in what way?

VERIFICATION (for verifiers)
1. How is the verification carried out?
a) What kind of verification do you perform?
b) How are you working with requirement management?
c) How is traceability handled in a project?

2. How to you think an agile way of working affects the verification?
a) Advantages?
b) Disadvantages?

DEVELOPMENT (for developers)
1. How is the software development performed?
a) Are you working with traceability in the projects and in what way? (Requirements, code).
b) How are you working with requirement management?

2. Do you participate in testing the code?
a) If yes, in what way?

3. How do you think an agile way of working affects the development of safety critical software?
a) Advantages?
b) Disadvantages?

COMMUNICATION
1. How do you communicate in a project?

2. Who are you communicating with?

3. In what way do you think an agile way of working affects the communication?

KNOWLEDGE/COMPETENCE
1. Do you feel that you have enough knowledge to work with safety critical application development?
a) What kind of education to you have within this field?
b) Would you like to have any education within this field?
If yes, what kind of education?
c) Do you have any other thought regarding competence?

GENERAL
1. What is your general opinion about working agile with safety critical application development?
a) What benefits are there?
b) What disadvantages are there?

OTHER
1. Anything else you would like to add?
a) Other parts of your work that could be relevant to discuss?

2. Do you know any other person that would be suitable to interview for this study?

3. Can I contact you if I would need to ask any questions?
[End recording]

Thank you for participating!

B.2 Swedish Interview Guide

INTERVJUGUIDE

Introduktion

[Börja ljudinspelning]

- Berätta lite allmänt om din yrkesroll och hur du arbetar.
  a) Hur ser en vanlig arbetsdag ut?
  b) Vad är din funktion i ett projekt?
  c) Hur länge har du arbetat inom din nuvarande roll?

- Vilken eller vilka agila metoder har du arbetat med vid säkerhetskritisk applikationsutveckling?
  - Hur länge har du arbetat agilt vid säkerhetskritisk applikationsutveckling?

- Arbetar ni efter EN50128 / IEC61508 eller andra standarder?

Huvudfrågor
DOKUMENTATION

1. Vilken typ av dokumentation utför du under ett projekt?
   a) Vilka fördelar/nackdelar tror du att ett agilt arbetssätt har när det kommer till dokumentation?

ORGANISATION
1. Finns det något organisatoriskt som skulle kunna göras för att underlätta ett agilt arbetssätt? (möten, uppbyggnad av arbetsgrupp etc.).

PLANERING
1. Hur ser planeringen ut i ett projekt?
   a) Vilka delar planeras inom din del av arbetet och på vilket sätt?

VERIFIERING (för verifierare)
1. Hur går verifieringen till?
   a) Vilken typ av verifiering utför du?
   b) Hur arbetar du med kravhantering?
   c) Hur hanteras spårbarhet inom projekt? (Krav etc).

2. Hur tror du att ett agilt arbetssätt påverkar verifieringen?
   a) Fördelar?
   b) Nackdelar?

UTVECKLING (för utvecklare)
1. Hur går utvecklingen av programvara till?
   a) Arbetar ni med spårbarhet inom projekt och på vilket sätt? (krav,kod)
   b) Hur arbetar du med kravhantering?

2. Deltar du vid testning av kod?
   a) Om ja, på vilket sätt?

3. Hur tror du att ett agilt arbetssätt påverkar utvecklingen av säkerhetskritisk programvara?
   a) Fördelar?

b) Nackdelar?

KOMMUNIKATION

1. Hur kommunicerar ni i projekten?

2. Vilka kommunicerar du med?

3. På vilket sätt tror du att ett agilt arbetssätt påverkar kommunikationen?

KUNSKAP/KOMPETENS

1. Känner du att du har tillräckligt med kunskap för att kunna arbeta agilt med säkerhetskritisk applikationsutveckling?
   a) Vilken typ av utbildning inom området har du?
   b) Skulle du önska någon ytterligare utbildning och i så fall vilken?
   c) Har du några andra tankar kring behov av kompetensutveckling?

ALLMÄNT

1. Vad är din generella uppfattning av att arbeta agilt vid säkerhetskritisk applikationsutveckling?
   a) Vilka fördelar finns?
   b) Vilka nackdelar finns?

2. Känner du att ett agilt arbetssätt och säkerhetskritisk applikationsutveckling motverkar varandra?

3. Finns det delar av projekt som du upplever skulle kunna förändras för att förenkla ditt arbete?
   a) Vilka är dessa delar och vad skulle kunna göras?

ÖVRIGT

1. Har du något som du vill tillägga?
   a) Andra delar av ditt arbete som kan vara relevanta att diskutera?

2. Känner du till andra personer som skulle kunna vara lämpliga för den här studien?

3. Finns det möjlighet att kontakta dig i efterhand jag skulle behöva ställa ytterligare frågor?

[Avsluta ljudinspelning]

Tack för att du ville medverka!

TRITA -EECS-EX-2018:755

www.kth.se