REQUIREMENTS ANALYSIS FOR AI SOLUTIONS

– A STUDY ON HOW REQUIREMENTS ANALYSIS IS EXECUTED WHEN DEVELOPING AI SOLUTIONS

Bachelor's Thesis in Informatics (15 credits)

Anton Olsson

Gustaf Joelsson

2019KANI28

UNIVERSITY OF BORÅS

Title: Requirements Analysis for AI solutions – a study on how requirements analysis is executed when developing AI solutions

Year: 2019

Author/s: Anton Olsson & Gustaf Joelsson

Supervisor: Bim Fagerström Kareld

Abstract

Requirements analysis is an essential part of the System Development Life Cycle (SDLC) in order to achieve success in a software development project. There are several methods, techniques and frameworks used when expressing, prioritizing and managing requirements in IT projects. It is widely established that it is difficult to determine requirements for traditional systems, so a question naturally arises on how the requirements analysis is executed as AI solutions (that even fewer individuals can grasp) are being developed. Little research has been made on how the vital requirements phase is executed during development of AI solutions.

This research aims to investigate the requirements analysis phase during the development of AI solutions. To explore this topic, an extensive literature review was made, and in order to collect new information, a number of interviews were performed with five suitable organizations (i.e, organizations that develop AI solutions).

The results from the research concludes that the requirements analysis does not differ between development of AI solutions in comparison to development of traditional systems. However, the research showed that there were some deviations that can be deemed to be particularly unique for the development of AI solutions that affects the requirements analysis. These are: (1) the need for an iterative and agile systems development process, with an associated iterative and agile requirements analysis, (2) the importance of having a large set of quality data, (3) the relative deprioritization of user involvement, and (4) the difficulty of establishing timeframe, results/feasibility and the behavior of the AI solution beforehand.

Keywords: Artificial Intelligence, AI, AI solution, AI system, Explainable AI, Requirements Analysis, Requirements Management, Requirement Engineering, Systems Development

Acknowledgements

We would like to express our sincere gratitude towards all of the participating organizations for this study. We would also like to give a special thanks to our mentor, Bim Fagerström Kareld (Senior Lecturer at the Faculty of Librarianship, Information, Education and IT (including The Swedish School of Library and Information Science)), for assisting us throughout our work.

Anton Olsson & Gustaf Joelsson

Table of Contents

1. Introduction

In this first chapter, the research area is presented. This is done via initially introducing an overlaying background to the research problem, followed by a problem discussion, which finally results in the purpose of the study as well as the research question. Lastly, a target audience is specified.

1.1 Background

As a result of the increasingly ubiquitous world we reside in, billions of devices contribute with endless streams of information every day. As a consequence, today's globalized business environment of is vastly influenced by the importance of implementing efficient and competitive technological solutions. Typically, these technological solutions have had the purpose of managing the streams of information in order to control and understand them, for example by turning raw streams of data into useful information that can be used for decision making within the organizations. This is achieved via, for example, implementing Information Systems (IS) within organizations, which provides business analysts with assayable information (Hoffer, George & Valachich, 2014).

However, the rapid pace of technological advancement has introduced a new kind of technological solution to the market; solutions that utilizes Artificial Intelligence (AI). AI is a vast and rather obscure subject that many individuals have a hard time to grasp, yet its importance for future business competitiveness is essential, and a growing number of businesses are faced with the challenge of developing and implementing AI solutions (Ghosh, 2018). AI solutions (or AI-enabled applications and systems) can briefly be described as machines (i.e, systems) that learn from experience, adjusts to new inputs, is able to make predictions and perform human-like tasks. This is achieved via Machine Learning (ML), which is a subset of AI that can be described as an automation of the analytical process, i.e, that the system can learn from the data, identify patterns and make decisions with minimal or no human intervention. In order for ML to perform such tasks, it uses Deep Learning (DL), which is a subset or type of ML that focuses on training the computer to learn on its own (Russell & Norvig, 2010).

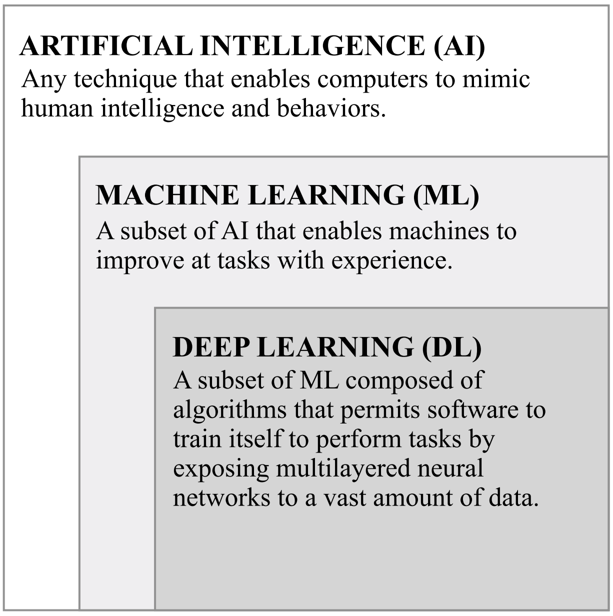

Fig. 1. An illustration visualizing the relation between AI, ML and DL (Russell & Norvig, 2010).

AI solutions are already adopted primarily within logistics, retail management, government surveillance and manufacturing assets management. AI is forecasted to penetrate more business processes in the future and it is estimated to become an increasingly important key feature in all future business models. In short; AI is expected to be present in everything digital (Ghosh, 2018).

However, the development of any variety of information systems is not an easy task; particularly not AI solutions. In order for organizations to develop any kind of system, a standard set of steps, called a systems development methodology, is used for support and development. A widely established methodology used to describe this process is the systems development life cycle (SDLC). It features several phases that mark the progress of the system- and analysis design effort. It is common that the amount of identifiable phases varies in the SDLC based on the authors perception of the methodology, but the general content, direction and ultimate goal is the same. However, despite the variation in phases, a widely recognized-, always included- and critical phase is the requirements analysis (RA) phase. In any systems development project, the RA is described as the foundation on which the success or failure of the project relies on (Hoffer, George & Valachich, 2014).

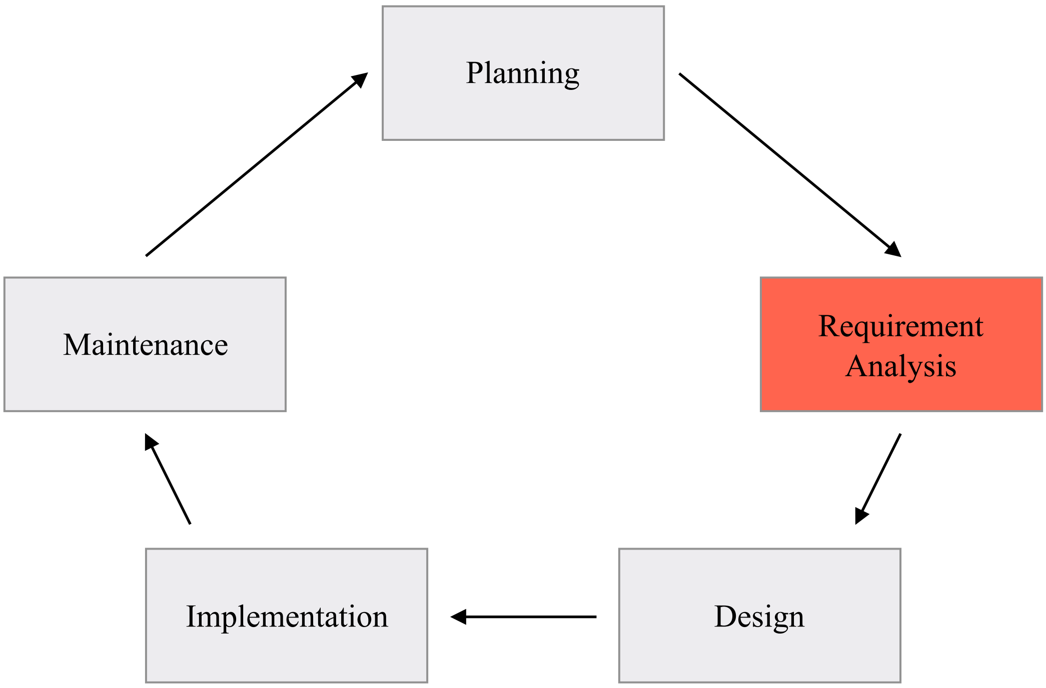

Fig. 2. An illustration visualizing the SDLC as described by Hoffer, George & Valachich (2014).

The SDLC visualized above is used to illustrate where the requirements analysis process is located in the SDLC. The goal of the RA is to determine what functions (i.e, requirements) the new system should inhabit. A requirement can be defined as a desirable function within the system. It might for example be functions that arose from an organizations desire to deal with problems in current procedures more efficiently or a desire for a system to perform additional tasks. Hence, during the RA, analysts work close together with the users to determine what the users want and need from the new system. The final step of the RA is to analyze interrelationships between the requirements in order to avoid redundancies (Hoffer, George & Valachich, 2014).

A poorly performed RA is often the greatest contributor to a failed project. If requirements are not well documented and set up in an efficient way, the final system will not inhabit the qualities needed from its users. Some of the consequences might be, for example, that the system development will be delayed, which in turn will make the whole project exceed the budget drastically. Furthermore, because of misjudged functional qualities, the system may risk to not perform and deliver in a satisfactory way for its users, which will lead to dissatisfaction (Eriksson, 2008).

1.2 Research Problem

Artificial Intelligence as a subject can be relatively difficult to grasp. With extensive increase in usage, interest and development of AI, numerous questions and potential dilemmas appears (Stone et al., 2016). One element in the subject of AI which is not well researched is the process of developing and implementing it as a solution into the corporate world. As with any systems development project, these AI solutions must follow a set of standardized systems development phases, such as the SDLC. This rises questions regarding how the development and implementations of AI solutions should be executed.

According to statistics in the CHAOS report from 2015, 29% of all software development projects were completed successfully, 19% were cancelled completely while 52% of projects were late and over budget. When determining the major contributors to successful projects, investments on user involvement (i.e, the part where requirements are determined by the users) was ranked in a shared first place (Hastie & Wojewoda, 2015). The statistics match theoretical assumptions on the importance of RA. For example, Eriksson (2008) describes that the cost of fixing a certain problem rises tenfold for each step in the SDLC process. A problem that costs 1000 USD to fix in the requirements phase will cost 10 000 USD in the design phase and 100 000 USD in the implementation phase, etc. Hence, It is widely recognized that the requirements analysis is a difficult yet acutely important phase for the success of any systems development project (Chakraborty, Kanti Baowaly, Arefin & Newaz Bahar, 2012).

When conducting requirements analysis it is usually done from the perspective of the users based on what functions they need and want. However, a reoccurring problem is that users often do not know what they want from the system, or if what they want is possible to implement in practice (Eriksson, 2008). Since it is already difficult for users to determine requirements for traditional systems, it arguably becomes even more challenging as they are faced with the implementation of obscure AI solutions. Not even multinational and widely established corporations with a lot of resources posses the capacity to accurately predict how AI solutions will behave.

A good example of this is Amazon. They attempted to develop an AI solution that could work as a recruitment tool to assist the company in finding the best suited people for certain job positions. Their AI solution worked by grading each applicants resume and giving them a score based on several aspects. The problem the developers found, was that the AI solution did not grade the applicants in a gender-neutral way, and rated male applicants higher than female ones. The AI solution based its grading on previous history regarding qualities by former Amazon employees from the past 10 years. Since most of the former employees were male, the system began teaching itself that male co-workers were more qualified than female ones. Although the AI was programmed without any bias in mind, it began to penalize resumes where words connected to the female gender was included (Dastin 2018).

All of the aspects lifted in throughout this chapter boils down to our research problem. Throughout this section it has been established that it is already difficult to determine requirements for traditional systems, so a question naturally arises on how requirements analysis is executed as AI solutions (that even fewer individuals can grasp) are being developed. Little research has been made on how the vital requirements phase is executed during development of AI solutions.

1.3 Purpose

The purpose of this work is to explore and describe how companies manage the requirement analysis phase during development and implementation of AI solutions. The relevance and need for this work is motivated by the lack of sufficient available research previously performed within this area.

1.4 Research Aim and Questions

As the interest of AI is increasing rapidly, more and more companies are leaning towards implementing AI solutions as a tool to increase their competitive advantage and promote their business (Stone et al., 2016). Implementing AI solutions must follow a set of systems development methodologies, such as the SDLC, where one of the key phases is the requirements analysis. This particular phase is vital to ensure the success of any system development project. Therefore, our aim is to research how this crucial phase is managed when developing AI solutions.

This leads to our research question:
- How is the requirements analysis phase executed during the development of AI solutions?

1.5 Target Audience

The target audience for this study are researchers and organizations who are going to perform / performing work within the fields of this study, i.e related to requirements analysis and AI. Organizations can use this study to broaden their understanding of the subject, while researchers can use the results from this study as a springboard for future research within the field.

2. Theory

In this second chapter, the theoretical framework for this work is presented. The purpose of this chapter is to provide an understanding of specific terms that are discussed within the subjects of AI, systems development and requirements analysis. Furthermore, the previous scientific works presented in this chapter will later be compared with the empirical collection in order to draw conclusions and provide an answer to the research question.

2.1 Information Systems

An Information System (IS) can be viewed in several ways and a number of definitions exist to describe the term as good as possible. The authors Laudon and Laudon (2014, p. 45) states that an IS can generally be viewed from both a technical perspective and from a business perspective, and gives the following definition of this kind of system:

> An information system can be defined technically as a set of interrelated components that collect (or retrieve), process, store, and distribute information to support decision making and control in an organization. In addition to supporting decision making, coordination, and control, information systems may also help managers and workers analyze problems, visualize complex subjects, and create new products.

In other words, an IS can be described as a system which has the ability to assist an organization in handling and presenting useful data as well as to assist the decision making within the firm. It translates raw obscure data into advantageous information that can be used by people within the company for making decisions and solving problems. The system can in the business perspective, assist the company in making better business decisions, and therefor helping the company to flourish (Laudon & Laudon, 2014).

From the beginning, information systems were uniquely developed for a certain type of organization, company or industry with a specific role and tasks. Such systems are called proprietary systems and are still widely used today. However, as information systems experienced a greater demand amongst several organizations, standardized information systems were created. Consequently, there are two major types of system of which are standardized systems and proprietary information systems. The standardized kind exist in a large variety for close to every industry and constitutes the foundation for many of todays businesses. A standardized system is developed to satisfy and comply with requirements from several companies within a specific industry. The goal with a standardized system is to make it possible for several organizations to use the same system, which in the majority of cases will make it much cheaper, easier to develop and easier to maintain. When implementing this kind of system it can in many cases work rather painless and without the need for changing it too much, but in most of todays cases when implementing a standardized system they need to be changed and adapted to the specific business (Hedman, Nilsson & Westelius, 2009).

2.2 Artificial Intelligence

2.2.1 Definition of AI

Artificial Intelligence (AI) is described in the most straightforward way as a computer with the capability to behave and operate in a manner that can generally be associated with an intelligent being, such as a human or an animal (Copeland, 2018). According to Russell and Norvig (2010), there are 4 different kinds of approaches within the subject of AI. These approaches have all the fundamental keynote that AI is technology that can imitate an intelligent living being. The four approaches are *thinking humanly, acting humanly* and *acting rationally*:

2.2.1.1 Thinking humanly

Thinking humanly, which is also called the cognitive modeling approach. Based on the keynote of analyzing the mind of a human being to understand how they think and to see the brain in action. When enough data is collected, it can be transformed into a computer program which will act accordingly. If the program matches the perceived behavior of the brain - it can be seen as evidence for the program to be able to operate in the human mind. This is proof of concept that AI is possible.

2.2.1.2 Acting humanly

This is based on a test proposal from 1950 called the Turing test. Basic explanation for this approach is that the computer, or with other words AI, acts in a way similar to a human. The test is based on if an interrogator who asks question by wiring them down - can get answers from a computer that him/her can not tell is from a computer. If the test was to be executed today, the computer would need to pass several tests such as to be able to communicate well (natural language processing), store received information (knowledge representation), use the stored information to answer questions (automated reasoning) and to be able to adapt itself to new environments and detect new patterns (machine learning). If the computer can pass all of these criteria - it can be called an AI.

2.2.1.3 Thinking rationally

Of which is also knows as the laws of thought approach, is the foundation for an AI to be able to think rationally - to tell what is correct and what is incorrect. For a computer to be able to act as a human - it need to be able to perceive informal information, sort it out for it to become formal and then make a decision accordingly to what it seem to be correct. One of the biggest challenges with this criteria for an AI - is for it to be able to ”think” the correct thing, and then act accordingly by for example solving a problem in practice.

2.2.1.4 Acting rationally

Refers to if a computer can execute operations rationally, based on the earlier mentioned rational thinking. A problem that easily occurs is that in many cases, a problem do not have a correct answer. This makes it difficult for a computer to handle, since it is made to be rational.

2.2.2 Machine Learning

Machine learning is a subset of the subject of AI (Hurwitz & Kirsch, 2018). The general definition and fundamental idea of machine learning is that a computer system is trained by a collection of examples to be able to perform certain assignments chosen by the user. Through machine learning, a computer can learn how to “act” and how to execute tasks it has not yet encountered based on the previous collection of examples. There are generally two major strategies that machine learning follows, these are supervised learning and unsupervised learning (Laudon & Laudon, 2016).

The first strategy, supervised learning, is based on the idea that the developer gives the computer system a collection of examples (a training set) which consists of data and the correct output. This can be compared to presenting a test to a student, but then also giving him/her the answer and telling the student how to use this information for solving similar problems or questions in the future (Laudon & Laudon, 2016).

The second strategy, unsupervised learning, is based on the idea that a computer system is trained by providing the training sets with useful data, but without the solutions or answers. This is more comparable to presenting a question to a student and telling them to find the pattern, for example which should be instead of the question mark in "1, 2, 3, ?, 5" - which is obviously 4 (Laudon & Laudon, 2016).

There are also subcategories to these major strategies of which one is reinforcement learning which is similar to the supervised strategy, but without the usage of a training set. This sub-sub-strategy focus on the technique of trial and error instead which makes every successful run useful to further reinforce the process. Another sub-category is neural networks and deep learning. This category is a newer one which have the possibility to provide a higher quality of machine learning (Hurwitz & Kirsch, 2018).

2.2.3 Deep Learning

Deep learning is a special strategy within the category of machine learning. It can be applied to both supervised and unsupervised learning approaches and the goal with most deep learning is to simulate the human brain to solve complex problems and issues, with a small amount of information. It has its foundation in the strategies of neural networks within machine learning, but on a much deeper layer (Hurwitz & Kirsch, 2018). Deep learning focus on machine learning with strategies based on computational models that are composed of several processing layers. It focus on these models to learn representations of data with numerous levels of abstraction (LeCun, Bengio & Hinton, 2015).

Deep learning discovers detailed structure in large data sets by using the back-propagation set of computer instructions to point how a machine should change its internal guidelines that are used to figure out/calculate the representation in each layer from the representation in the previous layer. It exist to make deep learning understand more profound meanings from less information (LeCun, Bengio & Hinton, 2015). Today, deep learning is used in a wide variety of industries, but with most success in for example voice recognition and Internet of Things (IoT[1]). It has the ability to for example predict when a specific machine will malfunction or how a fugitive will escape and therefor help law enforcement keep track of the movements made by him/her (Hurwitz & Kirsch, 2018).

2.3 AI solutions

2.3.1 Definition

There is no universally established definition of the term AI solution, but based on the foundation of the previous chapters that explained information systems and artificial intelligence, AI solutions is most easily defined as a system that utilizes AI to improve its

[1] "A vast network of physical objects with embedded microchips, sensors, and communications capabilities that link people, machines, and entire systems through the Internet" (Greengard, 2015).

functions. The purpose of developing and implementing AI is often to automize processes that previously required manual labour, as well as to assist in decision making. AI solutions are already widely used within businesses and they are a growing phenomena, which can be seen in a study from 2017 where 76% of the respondents regarded AI as a fundamental part to their business strategy. 64% of the respondents also believed that the growth of their business is dependent of a large-scale AI adoption (Infosys, 2016).

2.3.2 Examples of Successful AI solutions

Examples of usage for AI to promote business can be found in multiple different industries. American Express (AmEx) uses AI and machine learning to detect frauds in real-time which results in saving millions of dollars for both the company and their customers. AmEx also utilizes AI to tie customers purchases to specific products, which can create sets of data that can be useful for merchants to provide personalized advertising (Marr, 2018).

In previous chapter about deep learning, it is described that deep learning can predict when a specific machine will malfunction. This is exactly what the Swedish company Volvo is working on where they use AI to predict when a specific part in the car will most likely fail. They have also put a lot of effort into researching how an autonomous car should act in dangerous situations - all with the help of machine learning and AI (Marr, 2018).

2.3.3 Examples of Failed AI solutions

Amazon is one example that was also mentioned in the introduction. They created an AI system that accidentally sorted out women and gave them a lower score when trying to find the best candidates for hiring. It graded words as woman and female lower than words such as male. The faultiness was created because the dataset provided to the AI was not completely gender neutral - which resulted in that the system assumed that men were better because more men had been existing in the training datasets (Dastin, 2018).

Another great example which went very wrong was with the company Facebook and their development of AI chat-robots. Facebook launched a program in 2017 where they tried to create AI chat-robots that could argue and have a human-like debate. They used datasets as a foundation for the training of these chat-robots, and also made them talk to each other to further develop themselves. However, after some time, a problem came to the attention of the developers - the two chat-robots that had been talking to each other had started to shift from using English to using a new and own language, which they created by themselves. The language was more efficient than english for the AI robots, but their self developed language meant that the humans could not understand them anymore. The project was shut down and the reason behind all of it was set to be that the developers did not set a requirement that the AI:s should communicate in a specific language such as English (Nieva, 2017).

2.4 Explainable AI

As discussed in the previous chapter about AI (2.3), implementing AI solutions do not always provide the organizational benefits that is expected. There are increasing numbers of incidents reported in popular media that highlights AI systems making incorrect decisions, which in turn is accelerating an erosion of trust towards AI. According to Sheh and Monteath (2018), the rapid technological advancement and development of AI has come at the expense of the ability for humans to understand, verify and trace these intelligent systems. Sheh and Monteath (2018, p. 261) continues to explain that:

> [Organizations are beginning to] realize that in order for AI to be trusted, it must not only demonstrate good performance in its decision making, but it also must explain these decisions and convince us that it is making the decisions for the right reasons.

The lack of explainability is arguably one of the greatest flaws for current AI solutions. Not only does it impede users from realizing and defining requirements as the AI solution is being developed/implemented, it also contribute to immense difficulties for stakeholders ability to understand the system when it is up and running (Sheh & Monteath, 2018).

As a consequence of the obscurity surrounding AI, the topic of explainable AI (XAI) has become a popular topic over the past few years. According to Sheh and Monteath (2018, p. 261): ”XAI seeks to produce AI agents and systems that can not only make decisions but also appropriately satisfy the application’s requirements for associated explanations”. In other words, XAI seeks to be what its name infers - an explainable AI (Sheh & Monteath, 2018).

XAI is heavily reliant on explanatory requirements, which are requirements that describe how the AI is going to make itself explainable. These explanatory requirements are categorized into three dimensions; *source, depth* and *scope.* The *source* is where the explanatory information used for the decision comes from. *Depth* describes how attributes and models were used to make the decisions. *Scope* defines if the explanation is based on justification or teaching (i.e, if an AI makes a decision based on previous knowledge it is a justification explanation. On the contrary, if it makes a generalized decision outside of examples that have been seen, it is a teaching explanation) (Sheh & Monteath, 2018).

As AI becomes more involved in critical decision-making, the demand for reliable, transparent and accountable AI agents and systems/solutions are becoming a necessity. If the explanatory requirements (mentioned in the previous paragraph) are met, they will allow the AI to become explainable and fulfill the desirable properties of reliability, transparency and accountability. However, the trade-off of explainability is often occurring at the expense of factors such as performance and development efforts (Sheh & Monteath, 2018).

2.5 System Development Methodology

2.5.1 Definition

In order for organizations to develop any kind of information system a set of steps, called systems development methodology, are used for development (Hoffer, George & Valachich, 2014). Jayaratna (1994) estimates the amount of available methodologies to number around 1000. However, it is not easy to define exactly what a systems development methodology is. According to Huisman and Iivari (2006), there is no universally accepted, rigorous and concise definition of it. The term “methodology” is particularly frowned upon by the scientific community because it literary means a “science of methods”, while others argue that the term can be used interchangeably. This report will use Huisman and Iivari’s (2006) definition of systems development methodologies, which they define as a combination of:

1. **A system development approach**: The philosophical view on which the methodology is built. It includes a set of guiding principles, goals and beliefs, fundamental concepts and principles. For example, object oriented, structures and information modeling approaches.

2. **A systems development process model**: The representation of the sequence of stages through which the system is developed. For example, the linear life cycle model and the spiral model.

3. **A systems development method**: A systematic way of conducting at least one complete phase of systems development. This includes a set of activities, guidelines, techniques and tools based on a particular philosophy and the target system.

4. **A systems development technique**: Procedures to perform a development activity. For example, constructing entity relationship diagrams.

2.5.2 System Development Lifecycle

The systems development lifecycle (SDLC) is a traditional methodology for systems development that defines a detailed model on how to create, develop, implement, maintain and eventually fold a system. Namrata (2011, p. 261) describes: ”It [SDLC] is a complete plan outlining how the software will be born, raised and eventually be retired from its function”. This is achieved via a defining several phases that mark the progress of the system development. Hence, the SDLC provides an observable structure so that team members and project stakeholders are able to understand the current state of the system development project (Namrata, 2011). Furthermore, the SDLC is circular process. According to Hoffer, George and Valachich (2014, p. 33): ”The end of a useful life of one system leads to the beginning of another project that will develop a new version or replace an existing system altogether”. This is illustrated in the visualization below. The amount of individual phases differ based on the textbook author, ranging anywhere from 3 to almost 20 identifiable phases. For this report, the generic SDLC model that is presented by Hoffer, George and Valachich (2014) will be used. It consists of five major phases; *planning, requirement analysis, design, implementation and maintenance.*

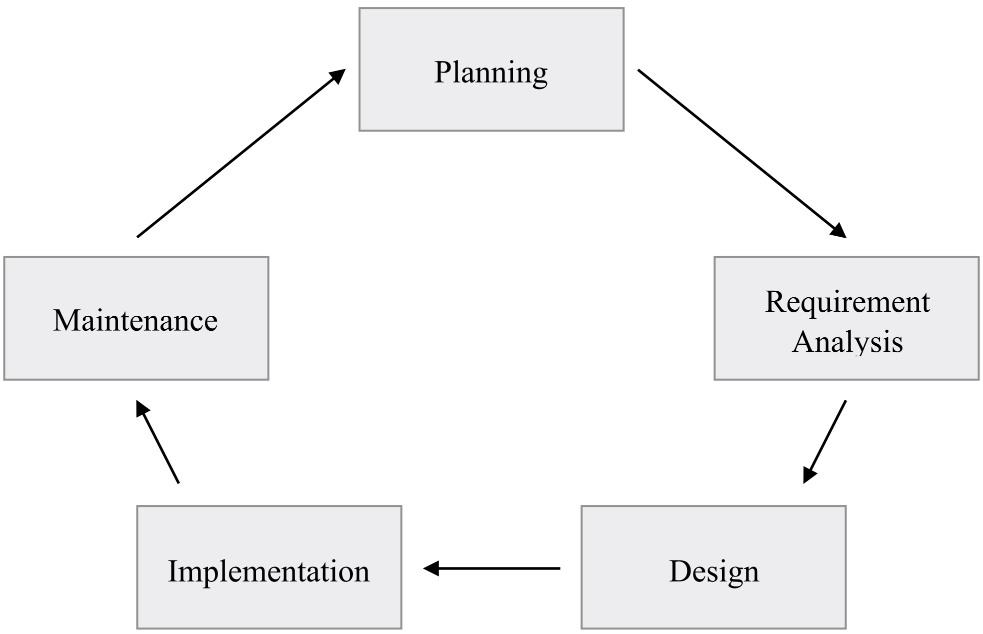

Fig. 3. An illustration visualizing the SDLC as described by Hoffer, George & Valachich (2014).

Below follows Hoffer, George and Valachich (2014, pp. 35-38) exact brief definitions of each process in the SDLC:

1. **Planning**: The first phase of the SDLC in which an organization's total information system needs are identified, analyzed, prioritized and arranged.

2. **Requirement Analysis**: The second phase of the SDLC in which system requirements are studied and structured.

3. **Design**: The third phase of the SDLC in which the description of the recommended solution is converted into logical and then physical system specifications. Design is further divided into two sub categories, logical and physical design
   - Logical Design: The part of the design phase of the SDLC in which all functional features of the system chosen for development in analysis are described independently of any computer platform.
   - Physical Design: The part of the design phase of the SDLC in which the logical specifications of the system from logical design are transformed into technical-specific details from which all programming and system construction can be accomplished.

4. **Implementation**: The fourth phase of the SDLC in which information system is coded, tested, installed and supported in the organization.

5. **Maintenance**: The final phase of the SDLC in which an information system is systematically repaired and improved. Layout, fonts, title, composition, size, etc.

2.6 Requirements Analysis Process

In the previous paragraphs, many various phases of the SDLC was presented. However, since this report focuses on the requirement analysis phase, this part seeks to describe this particular phase in depth, as well as to give a detailed description and definition of requirements.

2.6.1 Requirements

Requirements can be defined as desired functions and qualities in a system. They are expressed by an organizations stakeholders, for example its users, executives and reference groups. The requirements are collected by requirement analysts, who in turn mediates the requirements to the developers. Requirements can in turn be divided into different categories, these are; *functional requirements, non-functional requirements* and *normal, expected and sensational requirements* (Eriksson, 2008).

2.6.1.1 Functional Requirements

Functional requirements describes *what* the system is going to do, usually through functions that the system should be able to perform. In other words, many functional requirements are described through specifying expected in- and output data. For example, that a system should be able to create, save, change and delete customers data. However, even if all of the functional requirements are met, the system might still be perceived negatively by its users, for example if the response time is slow.

2.6.1.2 Non-functional Requirements

Non-functional requirements describes *how* a system should function. The non-functional requirements are vital for how the users will perceive the final system since these requirements for example describes usability and performance. For instance, if a system is alpha tested by 40 users but the final system is expected to handle 4000 users, it is critical that the system is developed to cope with the expected amount of final users from the start. Otherwise the final system will fail to meet the performance criteria as the amount of users increase from 40 to 4000. Therefore, it is very important to establish the non-functional requirements of a system from the beginning since they are arduous to add or alter later on.

Furthermore, Eriksson (2008) describes that the non-functional requirements can be divided into four sub categories, these are; *usability, reliability, performance* and *supportability*.

- **Usability:**
  The goal of *usability* is to make the system as user friendly as possible, i.e it should be easy for the final users to learn and remember how to use the system. The interface should be easily navigated; buttons, headings, error/help messages should be easy to understand. A common example of a usability requirement is that users should be able to perform tasks without using the mouse (for instance by navigating between buttons using only the tab key), or that new users should be able to understand and perform three tasks with an 80% success rate within their first 15 minutes of using the system.

- **Reliability:**
  The goal of *reliability* is to ensure that the system has as little downtime and errors as possible. In other words, an organization should be able to rely on the system. An example of a reliability requirement is that an error with medium severity should not occur more than three times per year, and the system should revert to normal functionality without the need of a system administrator. A common way to measure the reliability requirements is by using MTBF[2]. Furthermore, it is common to specify a requirement for how long the system is allowed to be inaccessible. For example, to specify that “the system needs to be accessible 99% of a year”, which equals a total acceptable downtime of 3.65 days per year.

- **Performance**:
  Performance is measured in response time, or the amount of transactions per time unit that the system is able to perform and how much time each transaction is allowed to take. A performance requirement can for example be to specify that if 400 users search and register errands simultaneously the systems response time is not allowed to exceed 3 seconds for each errand. In the previous paragraph it was mentioned that a system is expected to resume execution after a disruption, a common performance requirement in this scenario is to specify for how long the recovery is allowed to take, i.e the recovery time.

- **Supportability**: The aim of supportability is to develop a system that is easy and cheap to maintain. A supportability requirement could for example be that the system should be coded in line with the companies code standard and that everything should be well documented. This is because in-house expertise should be able to solve future system bugs

---

[2] Mean Time Between Failure (MTBF) is a measurement of the mean time between failures in a system (Eriksson, 2008).

as smoothly as possible. Supportability also aims to create user friendly error messages which are easily understandable.

2.6.1.3 Normal, expected and sensational requirements

Another classification of requirements are the normal, expected and sensational requirements. These are important for the overall perception of a system.

- **Normal** requirements are the most common requirements. They are usually expressed as requests and wishes by users, for example “I need a system that can save customers”. Hence, normal requirements are expressed by the users of a system. If they are met, the system will be perceived positively, and vice versa.

- **Expected** requirements are difficult to pinpoint. They can be defined as requirements that the users expects but do not express. For example, users might be familiar with saving new customers in the old system using only the F4 key. The users take this function for granted and forget to express it. When the new system is implemented the F4 saving function is not implemented, since the developers did not know about this expected function. This will lead to dissatisfaction. Hence, in order to find expected requirements, the developers need a broad knowledge of the users daily tasks, corporate routines, other systems within the corporation and the old system.

- **Sensational** requirements are most difficult to discover. These are requirements that are not expressed nor’ expected. An example of a sensational requirement is if the new system shortens a typical work task from five independent steps to three. If sensational requirements are not met it will not lead to dissatisfaction. However, if they are met, they will lead to an extreme satisfaction.

2.6.2 Determining and Structuring Requirements

Once a project has been initiated and planned, the processes of *determining* what the system should do begins. This is typically described as requirements determination. Throughout this stage, the goal is for analysts to gather as much information as possible on what the system should do from as many sources as possible. For example via interviewing and listening to users of the current system, observing users, reading reports, forms and procedures. In other words, determining requirements is similar to conducting an investigation (Hoffer, George & Valachich, 2014).

In order to perform a good requirements determination, Hoffer, George and Valachich (2014, p. 179) mentions five distinct characteristics that analysts must consider. These are:

1. **Impertinence**: Question everything!

2. **Impartiality**: Remain neutral and consider issues raised by all parties in order to find the best organizational solution.

3. **Relax Constraints**: Assume that everything is possible.

4. **Attention to details**: Every fact must fit with every other fact.

5. **Reframing**: Dare to look at the organization in new ways. I.e, consider how users view their requirements.

The outcome of requirement determination, if done correctly, is a vast amount of information. This is manifested via notes from observations, transcripts of interviews and analysis of reports, forms, etc. as well as other sources of information. To review and analyze such an extensive collection of information takes a lot of time and financial resources. Too much analysis is not productive, quite the contrary (Hoffer, George & Valachich, 2014).

After the requirements have been determined and the vast amount of information has been gathered, analysts enter the requirements *structuring* phase. This phase seeks to organize the information into meaningful representations of the current system and the requirements desired in the replacing system. This phase requires modeling of the processing logic and the structure of data within the system. This is typically done with using data flow diagrams (DFD[3]) (Hoffer, George & Valachich, 2014).

2.7 Methodologies and related Requirements Management Methods

In most cases the system development methodologies have their own view on requirement analysis and the requirements management methods differ a lot. It is also worth mentioning that several of the most common methodologies use techniques for the requirements that are not really included in the method, but which uses processes which are in line with the specific system development methodology (Eriksson, 2008).

2.7.1 Sequential Development

Sequential development means that the whole system is the developed and one single sequence indifference to other methods where the system is developed and delivered bit by bit (Eriksson, 2008).

The most common sequential development model is the waterfall model which was created in 1970. The method consist of several phases of the which are requirements analysis, design, coding, testing and implementation. The developer cannot move onto the next phase in the development until the last one is done. Figure 4 shows six different steps, which is the original look from 1970 - but has since the creation been adapted several times (Sommerville, 2011).

[3] "A diagram that depicts objects or processes and the possible flows of data among them is called a data flow diagram (DFD). DFDs are useful for depicting information about how an organization operates; the interfaces between an application and the people or other applications that use it; and the high−level design of an application" (Stevens, 2002).

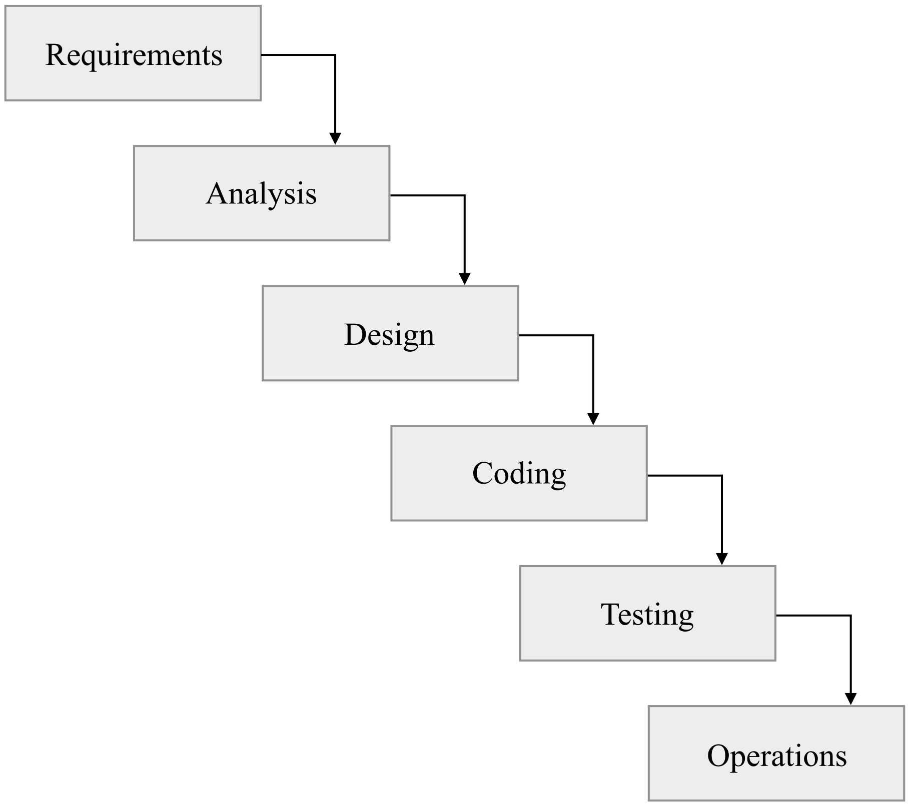

Fig. 4. An illustration visualizing the Waterfall Model as described by Royce (1970).

One of these modern adaption uses almost the same steps, but with different names and combinations. These steps are explained by Ian Sommerville (2011):

1. **Requirements analysis and definition:** This steps fundamental purpose is for the developer to establish goals, services and constraints of the system in development. All requirements are defined in detail and work as a system specification.

2. **System and software design:** This is the phase where the developer establish an overall system architecture. The design part involves identifying and describing the fundamental abstractions and their relationships in the developed system.

3. **Implementation and unit testing:** Software design is realized in this part of the waterfall model. In this stop the system has a set of programs or units. The unit testing refers to testing if all the previous specifications has been met.

4. **Integration and system testing:** All developed applications and programs are put together to form the final system. It is also tested to see that everything works together and that the earlier established software requirements are met. In the end of this step - the final system is delivered to the customer.

5. **Operation and maintenance:** Here the system is installed and maintained during the whole life cycle of the system. It involves fixing problems and errors which were not discovered earlier.

Before the entire project starts, the requirements for each step should be established. When leaving one phase for the next, it is difficult and can be very expensive to go back and make changes. The requirement phase will be frozen and cannot be changed. The steps above should never overlap - which means that the development should only focus on one phase at the time. One particular aspect to take into perspective in the waterfall model according to Balaji and Sundararajan Murugaiyan (2012) is that the the team who handles each phase, handles only that one and have no or very few contact with other teams. This may lead to costly problems in the end and that the system may be of bad structure when new requirements are discovered at a late stage. Some thoughts that arise here is why developers use this model if it is unchangeable and can lead to high costs when executed wrongly or poorly. There are several reasons: Requirements are set at an early stage which is easy to follow since no changes are made, it is very easy to implement since it is developed by a linear model, the amount of resources needed are very low and it consists of a high amount of documentation (Balaji & Sundararajan Murugaiyan (2012).

According to Ericsson (2008) the strive after perfection in each step of the waterfall model - may cause a lot of trouble because there are no real “perfect requirements” in real life. The longer time it takes to write and establish a perfect set of requirements, the higher the risk is that more requirements that are needed will be discovered and that the customer will therefore not be satisfied (Eriksson, 2008).

2.7.2 Iterative Development

In the sequential developing methods the system engineers tries to build an entire system all at once. According to Eriksson (2008) it is like trying to eat a whole elephant in one single meal. New kind of development methods has perspectives that differs from the traditional sequential development method. In more modern methods it is viewed as a better approach to dismember the elephant into small pieces and eat one piece at a time. This method is called iterative development (Eriksson, 2008).

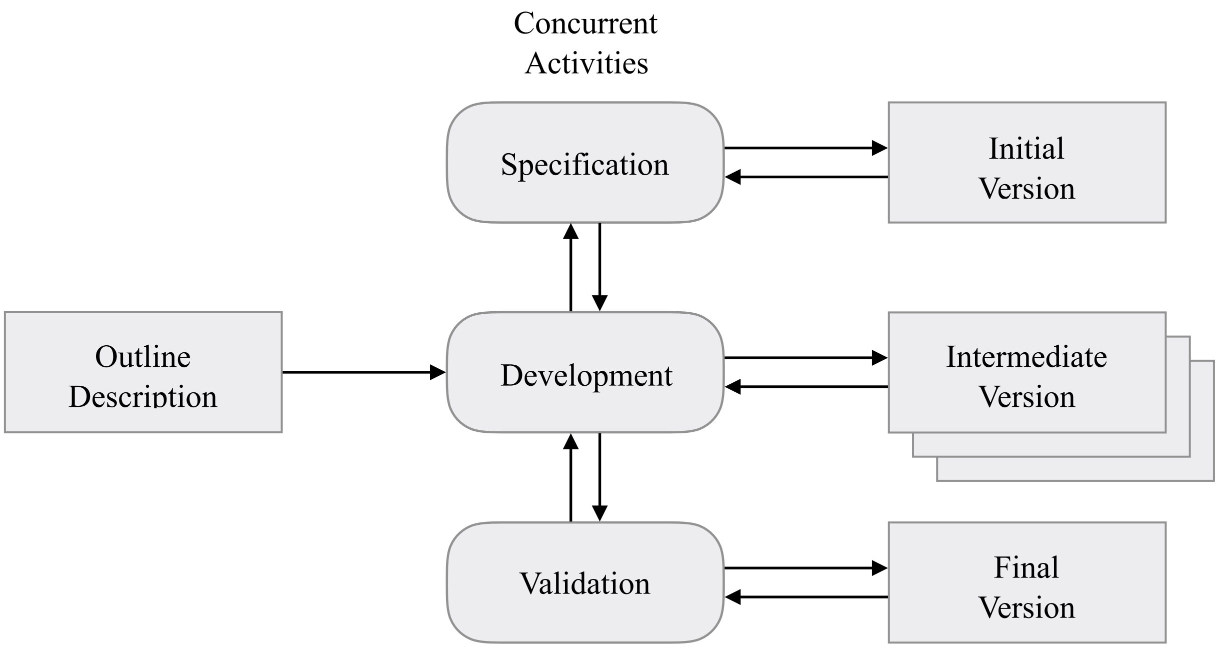

Fig. 5. An illustration visualizing the incremental model as described by Sommerville (2011).

Iterative development breaks the general project into a series of versions that are developed in a consecutive order. The most important and elementary requirements are bundled into the first version of the developed system. This version is developed quickly by a mini-waterfall method, and once implemented, the users can offer valuable feedback to the developers for it to be incorporated into the subsequent version of the system. Iterative development gets a preliminary version of the system to the users quickly so business value is provided. Since users are operating with the system, necessary further requirements could also be identified and incorporated into subsequent versions. The main disadvantage of iterative development is that users begin to work with a system that is by design incomplete and may consist of a lot of errors. Users should accept that solely the foremost essential requirements of the system are going to be accessible within the early stages and should be patient with the perennial introduction of new system versions (Dennis, Roth & Wixom, 2012).

For example: developers are building a system for sales storage and invoicing. In the waterfall model the whole system would be built immediately, but when using iterative method the different modules in the system will be developed with the essential functions at first. In later iterations of the development process the modules will be refined and more functionality will be added. As mentioned before, one of the biggest advantages with method kind of strategy is that it allows quick feedback between each iteration and it can therefore be more adjusted to the needs of the users and organization (Eriksson, 2008).

2.7.3 Incremental Development

The incremental development approach can be viewed as rather similar to the iterative one in many ways - and in several real-life cases they work together to form the perfect method. The incremental has its foundation in the idea that a system should be developed and early exposed to the customer for them to comment and give feedback for future versions. Activities such as validation, development and specification are interleaved instead of separate as it can be in other methodologies. The main core for our the incremental development is with other words - the importance of rapid feedback (Sommerville, 2011).

According to Sommerville (2011), incremental system development is an elementary part of the agile approaches, and it is superior to a waterfall approach for many companies, e-commerces, and personal systems. Incremental development reflects the approach that we tend to solve problems. We have a tendency to seldom work out a whole problem solution beforehand, but instead move toward an answer and solution in a series of steps, and backtracking once we understand that we have created an error. By developing the software system incrementally, it is cheaper and easier to produce changes within the software system as it is being developed (Sommerville, 2011).

Each increment or new version of the system incorporates a number of the functions that is required by the client or customer. In general, the first increments of the system embody the foremost vital or urgently needed functionalities. This implies that the client can measure the system at a comparatively early stage within the development to determine if it delivers what is needed and if the correct functionality is being developed. If not, then solely the latest increment must be modified and, possibly, new functionality outlined for later increments (Sommerville, 2011).

One of the biggest reasons for using the incremental approach is for the requirement phase. In a big system development project, requirements will always change a lot which can be very costly for the customers. In this approach - the requirements are allowed to change at every iteration, which can reduce the costs greatly when making changes to the system in development (Sommerville, 2011).

2.7.4 Agile Development

The agile development methodology relies on the incremental approach and therefor even the iterative. Beck et al. (2001) provides a description of the foundation of the agile way of thinking and developing:

> We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:
>
> > Individuals and interactions over processes and tools
> > Working software over comprehensive documentation
> > Customer collaboration over contract negotiation
> > Responding to change over following a plan
>
> That is, while there is value in the items on the right, we value the items on the left more.

The goal is to develop a high-quality system without the need of too much documentation. Tests of the system should be executed regularly without too much time in between and be developed in short iterations that are making the system bigger and better for each new iteration. There are several common development frameworks within the agile development environment, such as Scrum and DSDM (Dynamic System Development Method) (Eriksson, 2008).

A very popular way of dealing with requirements within the agile environment is through using a method called *MoSCoW (Must, Should, Could and Won't)*. This method is used to determine the importance of each requirement in the development process, which is done by comparing the requirements to a scale of importance. All requirements that are put under the *Must* part of the method must be fulfilled for the project to be viewed as successful. The *Should* demand should be fulfilled, but is not needed in a high degree for the project to be successfully completed. All of the *Could*-requirements can be left out and it will not change the outcome of the project at all. The last par of the method is the *Won't* part which are requirements that should not be realized now, but can be implemented in later stages of the development process (Eriksson 2008).

Within MoSCoW it is stated by Eriksson (2008) that each iteration should not consist of more than 60% Must-requirements with the reason that it may make it difficult for the developers to finish in time. If 40% or more of the requirements are non-must ones, the developers can decided to leave this out if problem with the Must-requirements should be detected (Eriksson, 2008).

2.7.6 Kanban

Kanban is a method used to improve the way a company can work and how they can develop further to be more efficient and therefor create a greater business value. The method puts a lot of effort to adjust itself into the organizations primary goals and to be ready for rapid change. A major aspect within Kanban is the usage of a delivery flow system which provides knowledge and limits the amount of *work in progress* by using visuals. These visuals prevents too much of a workload to be put on the workers, and also prevents the system from accepting to little work. When one task is completed - new are pulled in and becomes available. The Kanban method recognizes three agendas (Anderson & Carmichael, 2016):

- **The sustainable agenda**
  It is about determining a good pace of work and to improve focus. Kanban method uses this agenda to ensure not too much work is put on the workers at the same time. Big focus on the balance between demand and capability with both employees and customers in mind.

- **The service agenda**
  This almost declares itself. The service agenda's big focus is on providing the customers with the best possible service and giving them great satisfaction. This is seen as a key driver to success within Kanban, and many other methods as well.

- **The survivability agenda**
  This part exist for looking out for the future and to ensure that an organization survives in the future, even when change is happening. It is all about staying competitive and adaptive to new changes and new markets.

The Kanban method consist of six parts (Anderson & Carmichael, 2016):

1. Visualize
2. Limit work in progress
3. Manage flow
4. Make policies explicit
5. Implement feedback loops
6. Improve collaboratively, evolve experimentally

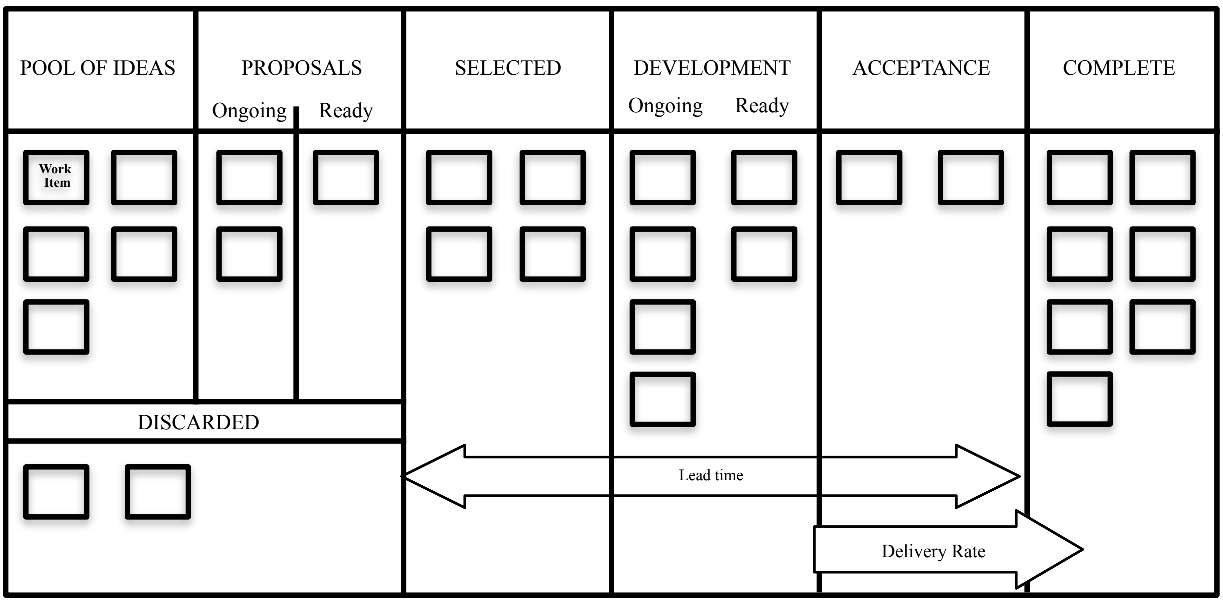

Fig. 6. An illustration visualizing the Kanban framework as described by Anderson and Carmichael (2016).

2.7.7 Scrum

Scrum is a framework that is used by software developers for achieving their goals with high quality and within the preset time. It is used when developing, implementing and maintaining complex products, and is most commonly used within software development. It is said by the creators of Scrum and the authors Schwaber and Sutherland (2017) that Scrum is not a technique or a method, but instead a framework which allows certain specific techniques, methods and processes to be used within it. The framework is made specifically to work well when using an iterative or incremental approach to the development which allows it to control upcoming risks and make predictions. There are three pillars within the Scrum framework that needs to be followed, these are *transparency*, *inspection* and *adaption* (Schwaber & Sutherland, 2017).

2.7.7.1 Transparency

Objects and aspects of significance must be viewable by all of the responsible parties. The transparency refers to that everyone involved in a project needs to have the same understanding of things being observed, such as that they should use the same definition for the word ”done” or that a common language should be shared by all involved individuals (Schwaber & Sutherland, 2017).

2.7.7.2 Inspection

This pillar of Scrum mainly refers to that the group of people working on a project must frequently check and inspect the processes and subtasks to see if unwanted variances has occurred. The purpose is to be sure the development is remaining on the correct track towards the set target (Schwaber & Sutherland, 2017).

2.7.7.3 Adaption

If the developers detect anomalies in the development process from the step of inspection, the adaption pillar refers to that they need to adapt and change in order to reach the final target. Changes or adjustments must be made closely following the detection of anomalies in order to prevent further deviation, so that the risk is for the project to go ”sideways” is minimized (Schwaber & Sutherland, 2017).

2.7.7.4 The Scrum Team

The first person that comes to mind when describing Scrum is the *Product Owner*. This person (never consists of more than one person) is mainly responsible for fully utilizing the work of the entire Scrum team. He/she is responsible for handling the Scrum Backlog as well (which will be describe later on). Managing the backlog for the product owner refers to expressing backlog items, prioritizing the backlog items and that the backlog list is clear, transparent and understandable for every involved entity (Schwaber & Sutherland, 2017).

The second group of people of the Scrum team is the *Development Team*. Their primary tasks is to finish the requirements and tasks which is set in the backlog by the product owner. The development team do not use roles or titles. Their goal is to complete the backlog items and move them to the ”done” section within each iteration (Schwaber & Sutherland, 2017).

Last person to describe in a fully functional Scrum team is the *Scrum Master*. This roles almost describes itself - the Scrum masters primary task is to promote Scrum within the team, and can be viewed as the leader of the team (Schwaber & Sutherland, 2017).

2.7.7.5 The Scrum Events

The first important event of the Scrum framework is called the *Sprint*. This is another word for every iteration of the development. The sprints are a maximum of one month, but can often be less and it is not possible to change the time when the sprint has already begun. No changes can be done to the project that could change the sprint goal. The only time the sprint can be changed, is if it is canceled for the reason that it is impossible to continue on the same path for the whole sprint period (Schwaber & Sutherland, 2017).

Before each sprint the team should conduct something called *Sprint Planning*. The aspects to take in consideration here is what can be delivered in the next sprint, and how will the team achieve the sprint goal (Schwaber & Sutherland, 2017).

Daily Scrum is the replacement for meetings inside the Scrum framework. It consist of a 15 minute daily meeting where the team members tell each other what they have done, if there was any problem and what they will do today. If problems occur, other team members should assist into solving those problems (Schwaber & Sutherland, 2017).

After each sprint there is a term called *Sprint Review*. This exists for the Scrum team to explain what was achieved during the last sprint, what can be improved to next sprint and review the status of the whole project (Schwaber & Sutherland, 2017).

Sprint Retrospective exists for the Scrum Team to inspect itself and see what can be improved for upcoming sprint (Schwaber & Sutherland, 2017).

2.7.7.6 Scrum Artifacts

Product Backlog is the first artifact of Scrum. The backlog is a term used in Scrum with the meaning of a sorted list consisting of all requirements and fixes that needs to be done in order to reach the sprint goal and the final goal. A backlog in Scrum is a dynamic artifact and can never be called completed. It is described that the requirements never stop changing, which makes the product backlog a living entity. An area specific rule used in the product backlog is that the higher order on the list, the more detailed objects you can find (Schwaber & Sutherland, 2017).

Sprint Backlog is the second artifact of Scrum. This artifact is most easily described as specific items from the product backlog that is chosen to be executed within a certain sprint. The goal is that the development team should be able to mark the chosen item as ”done” in the end of that sprint. The sprint backlog is adjusted according to the estimated remaining work required (Schwaber & Sutherland, 2017).

3. Methodology

This third chapter provides a description of how the empirical data was gathered and managed for this work. It also provides arguments for why a certain approach was used.

3.1 Choice of Method

According to Patel and Davidsson (2003) there are two main research methods available when collecting data; qualitative and quantitative methods. The quantitative approach includes collecting data that is highly structured where results are quantifiable and statistical. The premiss of the quantitative method is that reality is objective and that it is possible to measure the available information (Patel & Davidsson, 2003). In other words, this approach gathers data through for example numerical studies, statistical gatherings and questionnaires with simple and quantifiable answers (Creswell, 2002). On the contrary, the qualitative approach includes collection of data with a low structure. This approach focuses on deeper understandings of concepts as well as emphasis on details and descriptions rather than units of measure (Patel & Davidsson, 2003). I.e it is based on the assumption that reality is far too complex to be reduced into numbers, and instead, one must seek to gather information in the form of *words,* which provides the opportunity of more nuanced and elaborated answers (Jacobsen, 2017). Thus, the data is typically collected via the use of interviews or case studies in order to provide the information characteristics associated with the qualitative approach (Patel & Davidsson, 2003).

This work seeks to answer a complex and intricate question. The knowledge needed to answer the research question is therefor not of a quantitative character, nor' is the purpose of this work to provide a quantifiable answer. Instead, it is of great importance that the information used to create the result is nuanced, deep and rich in its character. Through profound interviews with companies we want to achieve answers that are able to be analysed and provide us with the deep understanding and knowledge that we need for our result. Therefore, our choice of method is the qualitative method.

The qualitative procedure puts its emphasis on individuals, for example the interviewee in an interview, and on his or hers perception and knowledge of a certain subject. In other words, the qualitative method facilitates the collection of data by providing the opportunity to interview professionals who are active within the field of interest (Patel & Davidsson, 2003). Hence, we have chosen to interview individuals who are knowledgeable and experienced in the field we are going to research.

Furthermore, our choice of method can be tied to the comparative research method. Bryman and Bell (2013) defines this method as one where two or more subjects are studied through identical methods, for example via interviews. In other words, it can be described as a method where the researcher, through comparisons of results, is able to achieve interpretations and opinions that are equal (as well as different) among subjects who have experienced similar events in similar contexts (The SAGE encyclopedia of qualitative research methods Vol. 1. 2008). In the case of this work, that can be translated to individuals who are experienced with the requirements analysis phase as organizations developed and/or implemented AI solutions. As described in the previous paragraph, our chosen interviewee's are individuals who fit those criteria. For this work, the perspective of five different organization has been chosen. The benefit of not including more organizations is that each respective organization can be profoundly examined, which will contribute to rich, deep and nuanced information that is vital for making relevant conclusions. However, since the amount of five participating organizations is relatively small, the conclusions in this work can not be viewed as fully universally representative (Bryman & Bell, 2013).

3.2 Approach

There are three major reasonings that can be applied when relating theory and empirics to each other. These are *deductive, inductive* and *abductive* reasonings. According to Jacobsen (2017), the *deductive* (also; top-down) premiss is that the empirical collection is based on available theory and conventional principles. In other words, the available theory decides which type of information that should be collected, how this information should be interpreted and how the result should be related to the theory (Patel & Davidsson, 2003). I.e, the process moves from theory to empiricism. This requires the researcher to have an established theory before collecting data. In contrast, when the *inductive* (also; bottom-up) reasoning is used, the theory is formed as a result of observing reality (empirics). In other words, the empirics form reality based theories, i.e the process moves from empiricism to theory. Therefore, it is of great importance that the researcher keeps an open mind when collecting data with an inductive reasoning (Jacobsen, 2017).

However, Jacobsen (2017) argues that from a pragmatic perspective, it is impossible to remain purely deductive or inductive. Instead, research is often a certain mix that contains both reasonings. Such perspective is called *abductive* reasoning. Its premiss is that all research begins by an observation of a surprising phenomena, which generates a question. The question is in turn regarded as a problem that has to be solved, which generates speculations as to why things are as they are. The speculations are then generated into hypotheses and assumptions, which naturally leads to an empirical investigation to see if the hypotheses and assumptions are correct. In short, research becomes a continuous problem solving process, which makes it a combination of both deductive and inductive reasonings. I.e, abductive reasoning is a constant interaction between theory and empirics, where no one has a higher priority than the other (Jacobsen, 2017).

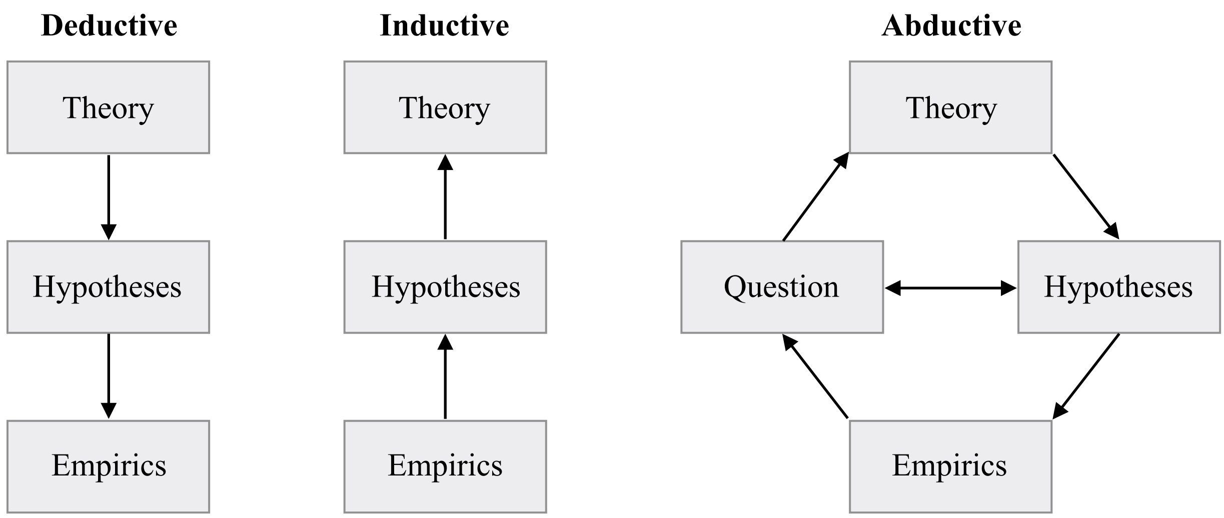

Fig. 4. A visualization of the differences between deductive, inductive and abductive reasonings (Jacobsen, 2017)

In this work, the relation between theory and empiricism is characterized by an abductive reasoning. While it is true that the majority of our process for this work coincides with the deductive approach (i.e we started with an extensive literature review that resulted in a theoretical framework on which we based our empirical collection), our process has also been influenced by the inductive approach. For example, as we conducted our empirical gathering we were presented with previously unknown information that influenced and further developed our theoretical framework. Thus, our process started by the theory moving towards empiricism, then back to theory from empiricism, etc. I.e, it has been a combination of inductive and deductive reasonings, which is the motivation behind why our reasoning coincides with the abductive approach.

3.3 Data Collection

The data collection was done with two goals in mind. The first one was to perform an extensive literature review with the goal of providing this work with a scientifically accurate and credible theoretical framework. This was achieved via compiling previously performed scientific works from other researchers, which in turn created a cohesive literary that covered various important aspects from areas related to our research. The second intention was to gathered information from an empirical point of view with the goal of providing this work with information from professionals within the field. In our case, this equaled to performing qualitative interviews with individuals within organizations that had experience and knowledge with development and/or implementing AI solutions.

3.3.1 Literature Review

The literature review was performed via the gathering of secondary sources consisting of previous scientific work within our research area. Patel and Davisson (2003) describes the literature review as document studies, where existing data and their appurtenant conclusions and results are reviewed. It can be explained as researchers seeking the answer and purpose of their study in previous studies with the same theme. Furthermore, the purpose of the literature review is to create a broad picture of the research field as well as to supply scientific knowledge to the study. Patel and Davidsson (2003) emphasizes the importance of source criticism as well as a variety of secondary sources. This is to achieve a variety of credible perspectives.

The secondary sources used in this study were gathered via the university of Borås library, particularly via their "Primo" search function. The selection of secondary sources have been made with the criteria mentioned by Patel and Davidsson (2003) (as mentioned in the previous paragraph) in mind. However, since our research topic is relatively new, we have had to reside to a few non-scientific sources to collect information about, for example, examples of AI solutions and failed AI solutions. Such non-scientific sources have been carefully investigated to ensure the highest possible credibility. For example, via using news articles written for well known and credible news papers (such as Forbes).

Keywords: Requirements Analysis, AI, AI solutions, Systems Development, XAI, Requirements Management

3.3.2 Selection

As stated by Bryman and Bell (2013) there are three main categories of selection. These are network-, strategic- and comfort selection. The comfort selection is based on that the choice of interviewee is made purely on if the first person asked for the interview will say yes. Even though this form of selection is based to a high grade of randomly finding people, it is often combined by making some kind of restriction such as age or which kind of company the interviewee work at. The strategic selection is based on if the interviewee fulfills certain criteria such as work position, knowledge or education. The last form of selection is the network-selection which is based on finding interviewees by tips and connections that you already have from prior interviews.

For finding the perfect candidates for this particular study, the selection process has been rather strict and in accordance with the principles of Bryman and Bell (2013). Therefore, we chose to use the strategic selection process from an organizational perspective. We wanted to find companies that had different goals and business interests. According to Jacobsen (2017), this creates a good foundation to get impartial, trustful and valid information. We only contacted organizations that outspokenly developed AI solutions. This is motivated by the fact that such organizations would be suitable for this study since they possess knowledge within our research area, it also facilitated our search for suitable organizations. To conclude, respondents have been selected based on their knowledge about AI solutions development. The final number of respondents for this work amounted to five organizations. The amount of participants varied, and ranged from 1-2 individuals in each interview.

3.3.3 Interviews

According to Bryman and Bell (2013), the most common data gathering method used when conducting qualitative research is interviews. According to Jacobsen (2017) there are three major distinct forms of interviews for qualitative research. These are unstructured interviews, semi-structured interviews and highly structured interviews. The level of structure refers to which degree the interview is following a set of predetermined questions (i.e, the questionnaire), the openness or closeness of the questions, as well as how the interview is structured. For example, when conducting an unstructured interview, there is a lack of structure and predetermined questions. On the contrary, when conducting a highly structured interview, there is a high degree of structure and predetermined questions. Highly structured interviews usually borderline to a quantitative questionnaire, while unstructured interviews share characteristics with a relaxed and spontaneous conversation. Both of these interview methods have several negative aspects. For highly structured interviews, the most prominent negative aspect is the suffocating nature of the interview, i.e. that the respondent is not able to speak freely about a certain topic, which may result in less nuanced information. For unstructured interviews, it is rather the great amount of openness that contributes to the possible negative aspects, since the lack of an interview guide entails a risks for the interviewers to forget certain important themes that needed to be discussed (Jacobsen, 2017).

Consequently, semi-structured interviews have been chosen as the interview method for this work. Bryman and Bell (2013) describes semi-structured interviews as a method where the interview has a list with predetermined questions and themes to discuss during the interview, but that this list should be regarded as a guideline rather than a strict order. This equips semi-structured interviews with the ability to provide a sufficient structure to maintain important themes while also offering the researcher an ability to ask for complementary questions and clarifications. It also enables the respondent to bring up themes (Jacobsen, 2017). Robson (2016) notes that it is common for the order and formulations of the questions to change based on the flow of the interview. This leads to unplanned questions and a chance to follow up on interesting and unclear aspects, which ultimately provides a good foundation for achieving nuanced and rich information.

The execution of the interviews in this work were performed based on our (as well as the interviewee's) disposable timeframe. We opted for interviews where it was possible to meet the interviewee face to face. However, since such interviews demand a lot of resources, we had to reside to the best suitable means for each particular situation. As a consequence, two out of our five interviews were done face to face, while the remaining two were done via voice over internet protocol (VOIP), i.e a telephone service over the internet. The benefits of face to face interviews are that they establish a high sense of trust and openness while simultaneously distancing the interview from any distractions. Negative aspects cover the difficulty of arranging meetings, based on geographical as well as time related aspects, it is also very costly. This is why VOIP was used for three interviews. The benefits of such interviews are that they are easier to perform and more cost effective, while simultaneously emphasizing the focus on the conversation. The negative aspects of VOIP interviews are that they do not establish the same sense of trust and openness as face to face interviews, since there is no physical interaction. Furthermore, some of the interviews were complimented with followup questions via e-mail. The benefits of E-mail is that it is very cheap and cost effective. Negative aspects are that it might contain less nuanced answers with a slow response time (Jacobsen, 2017). This is why they only were used for smaller followup questions.

3.4 Analysis

According to Alshenqeeti (2014), a considerable con with the qualitative approach is the high level of complexity with the information gathered during interviews. This is because of the nuanced, unstructured and vast amount of information that qualitative interviews generally result in. To combat this, researchers use different models to analyze the information efficiently. For this work, we have used the hermeneutical spiral model described by Jacobsen (2017). The model consists of four stages; *documentation, explore, systemize and categorize* and *Concatenate*.

1. **Documentation:** This first step follows directly after the interview. The goal of the documentation stage is to transcribe the interview, which creates a foundation for the next stage.

2. **Explore**: During this stage, the goal is to explore the transcribed data and search for relations. This is done relatively unsystematically. When this is completed, the researchers should have achieved a general overview as well as a few abstract themes.

3. **Systematize and categorize**: The goal in this stage is to reduce incalculable information. The text is then divided into different categories, which can be based on themes, places, events, etc. In other words, the information is categorized based on criteria that the

researchers decide. The information that can not be categorized is reduced. As this stage is completed, the researchers should have a graspable information collection.

4. **Concatenate:** In this last stage, the goal is to concatenate categories that share resemblances. For example, a collection of categories might highlight the same overall theme. Consequently, they are concatenated. By doing this, the researchers can group categories that are related, which in turn makes the information more concrete.

3.5 Reliability, Validity and Generalization

Reliability and validity are key aspects when determining trustworthiness of any study. Throughout this work, measures to achieve a high standard of reliability and validity have continuously been taken to ensure a high quality. According to Bryman and Bell (2013), the goal of reliability is to ensure that the study is as close to a universal truth as possible. In other words, it should be possible for independent parties to perform the same study several times and achieve in the same conclusion. Reliability seeks to ensure that the result can be independently reproduced and that the study has been conducted through reliable methods (Bryman & Bell, 2013). According to Jacobsen (2017), the reliability of a study that uses the qualitative method (which this study does) is closely connected to how data has been collected, as well as how it has been processed. It is, for example, important for the researcher to establish if the collection of data (and the context in which it has been collected) has affected the final conclusions and results in any way. To ensure that the reliability of this work is as high as possible, several different sources have been used in order to collect varied information that highlights the same field (in this case the requirements analysis phase when implementing AI solutions) through different perspectives. By doing so, the quality of this study is strengthened.

According to Bryman and Bell (2013), validity can be described with a question regarding wether the study observes, measures or identifies what it set out to do. In other words, to which degree the study is relevant in regards to the research purpose. To ensure a high validity, this work has developed a logical connection between the research question and the purpose with this study. Furthermore, the literature has been chosen based on the purpose and problem discussion.

Moreover, validity is often separated into internal and external validity. According to Jacobsen (2017), internal validity seeks to establish wether the results can be perceived as correct. In other words, if reality matches the researchers description of reality. There are three major aspects that must be taken into account for this:

1. Has the object of study given a true description of reality?

2. Has the researcher interpreted the description correctly?

3. Do the researchers results and conclusions reflect reality?
(Jacobsen, 2017).

According to Cohen, Manion and Morrison (2000) it is impossible to achieve complete validity, instead, one should try to achieve a certain level of it. For example, to establish a high internal validity for this work, we have answered the first of Jacobsen's (2017) question

((1) has the object of study given a true description of reality?) by interviewing organizations that have a lot of experience within the field we are researching. Furthermore, our interviewee's were chosen based on their particularly relevant skillsets within the organization. By doing this, our objects of study are as trustworthy and credential as possible, which constitutes a good foundation for achieving a correct description of reality.

The purpose of external validity is to establish to which extend the result of the study is generalizable. In other words, if the results of the study can be said to constitute general or universal results. Researchers who use a qualitative method is often limited by the amount of participating study objects (Bryman & Bell, 2013), and this study is no exception. This study performed interviews with five organizations. Hence, the results in this study is arguably only generalizable among the included organizations, but may not be regarded as universally representative.

3.6 Ethical Aspects

Bryman (2018) describes four ethical principles that all Swedish studies should fullfil. These are principles regarding; *information, consent, usefulness* and *confidentiality*.

1. **Information**: This principle serves the function of informing included individuals about the purpose of the study. During the empirical collection for this work, all included individuals (i.e, primarily the interviewees) were briefed about the purpose of this work as they were first contacted, as well as during the first minutes of the interviews.

2. **Consent**: The principe of consent establishes the aspect of voluntary participation. All of the included individuals for this work were asked to participate and did so at their own terms and will. Furthermore, they were also asked if they agreed to the interview being recorded, in order for them to achieve full control of their participation.

3. **Usefulness**: The principle of usefulness advocates that the information collected about the included individuals only are used for the context of the research. For this work, this principle has been followed strictly.

4. **Confidentiality**: This principle seeks to establish that the included individuals, and the information they provide, are treated with the highest level of desired confidentiality. This is why the interviewees have been anonymized in the empirical presentation, and why the interview data is not published as an attachment in this document. However, in the Empirics chapter, the individuals title and relevant education is presented, as well as the name of the organization. This is done to provide the readers with an understanding of the competences inhabited by the included individuals, while simultaneously not compromising the individuals identity.

4. Empirics

In this fourth chapter the results of the interviews are compiled and presented. The chapter is initiated with a short presentation and description of the participating companies and interviewees (in alphabetical order), which is followed by relevant results from the interviews. The names of the interviewees have been anonymized, but their title and relevant education is presented. The results that are presented are based on the different themes and categories that was discovered during the analysis, as described in the methodologies chapter. These themes are categorized and presented based on which overlaying theoretical field they are connected with.

4.1 Company Descriptions

4.1.1 Centiro

Centiro is a leading innovator of cloud services for helping customers to manage and control transportation and supply chain activities in the e-commerce, logistics and industrial sectors. Centiro's solutions are used by world leading companies and brands in more than 125 countries across the world. The company was founded in 1998 by the current CEO Niklas Hedin (Centiro, 2019).

The interview was conducted with two employees, below follows their title and relevant education:

1. Subject 1: Chief Data Scientist at Centiro: Bachelor of Science (B.Sc) in Computer Engineering, Master of Science (M.Sc) in Computer Science with a focus on Artificial Intelligence.

2. Subject 2: Delivery Manager at Centiro: Master of Science (M.Sc) with a major in Informatics.

4.1.2 Evry

Evry is one of the leading IT services and software providers in the nordic region. They develop and provide industry-leading solutions that give their customers a digital advantage, modernize business processes and make IT operating services more efficient. Evry was founded in 2010 as a merger between the two largest IT companies in Norway, EDB and ErgoGroup, together with the Swedish IT company Systeam (Evry, 2019).

The interview was conducted with one employee, below follows their title and relevant education:

1. Subject 3: Chief Architect and Advisor at Evry: Master of Science (M.Sc) Electronics.

4.1.3 Findwise

Findwise is a global expert in solving information challenges. The company builds search driven applications to help customers find, analyze and act upon information. Their award winning solutions are used by industry giants like SKF, Scania, Vattenfall, together with over 450 other clients worldwide. The company was founded in 2004, and is now a member of the Evry Group (Findwise, 2019)

The interview was conducted with one employee, below follows their title and relevant education:

1. Subject 4: Business Area Manager - Cognitive Solutions and AI at Findwise: Bachelor of Science (B.Sc) with a major in Cognitive Science, Master of Science (M.Sc) in Informatics and Business Technology.

4.1.4 Lexplore

Lexplore is a startup that charts how pupils read by utilizing eye tracking and artificial intelligence. The purpose is to achieve an objective overview of a pupils reading ability in order to aid and assist pupils with reading difficulties in time, as well as to promote and facilitate reading for everyone. The startup is now active in Sweden, The United Kingdom as well as in The United States of America (Lexplore, 2019).

The interview was conducted with the two founders, below follows their title and relevant education:

1. Subject 5: Co-founder of Lexplore and Researcher at Karolinska Institutet: Master of Arts (MA) in Computational Linguistics, Doctor of Philosophy (PhD) in Computational Linguistics.

2. Subject 6: Co-founder of Lexplore and Researcher at Karolinska Institutet: Master of Arts (MA) in Computational Linguistics, Doctor of Philosophy (PhD) in Computational Linguistics.

4.1.5 Swedavia

Swedavia is a fully state-owned company that owns, operates and develops the main network of Swedish airports. The company control and and owns 10 airports around Sweden. Their idea and vision is to provide a smooth travel experience for their customers as well as promoting a sustainable growth for Sweden (Swedavia, 2019).

The interview was conducted with two employees, bellow follows their title and relevant education:

1. Subject 7: BI-architect at Swedavia within the group of Finance, Purchasing & IT: Higher education within the area of economics.

2. Subject 8: Requirement Lead at Swedavia within the group of Finance, Purchasing & IT: Bachelor of Science (B.Sc) in Business Administration.

4.2 Collected Empirical Data

4.2.1 Descriptions of AI solutions

4.2.1.1 Purpose

All of the organizations that were interviewed develop and/or implement AI solutions for the purpose of automatizing and/or complementing manual labour. This is done primarily in the interest of their customers and also in some cases even for their own operations. In other words, the common ambition with the development of AI among the interviewees is that they want to eradicate monotonous and mundane tasks, for the purpose of making their system and even their business more efficient with less required manual effort. As an example, subject 3 at Evry states that almost every chat-robot online exist with the goal to reduce human input by being able to answer the first two questions unassisted by people. Swedavia continues by stating that they develop and use AI for the purpose of putting human focus on more important things and to be able to analyze data in a way that would not even be possible to do manually. To clarify, the purpose is to make system processes more streamlined and efficient, which allows organizations to redirect human endeavor on more important tasks.

4.2.1.2 Approach

A common theme amongst the participants was that every organization emphasized that AI solutions, or what some call AI systems, are larger systems that utilizes an/several AI function(s). In other words, the terms AI system and AI solution are interchangeable, since the AI solution in itself is just a small part of a larger system. Subject 1 from Centiro gave the following explanation:

> Let's say you have a car, right, but you might have a car with an AI system in it. The product is still the car. The product is not the AI itself.… It does not matter if you call it an AI solution or AI system, because at the end of the day it is a system that utilizes AI.

A mutual conclusion among all of the participants was that the term AI was not typically used, instead, all of the participants preferred to refer to it as Machine Learning (ML). This was motivated by that the term AI has a risk of referring to the wrong area within AI, and that machine learning is a more correct way to adress their actual operations and development. According to Centiro, ML is the most interesting subfield of AI these days and it is also where the organizations' focus is. When speaking of supervised and unsupervised learning (two common concepts in ML), the majority of companies in this study work primarily with the supervised method. This provides them with resources to teach a machine to do something specific. The only exception is that one organization states that it differs from project to project and that no general perception can be given.

4.2.1.3 Types of AI Projects

From the interviews it was understood that there are several types of machine learning development projects. The two main ones covered in our empirical gathering are categorized into standardized AI solutions (which requires no development of the AI solution itself, instead it is much like purchasing a standardized information system) and pure development projects (where the AI solution is developed from scratch). According to subject 3, purchasing a standardized AI solution and implementing it in organizations is the most common form of AI solutions projects. This has to do with the resource-intensive nature associated with development projects. Purchasing a standardized AI solution can be resembled by purchasing a certain function, for example image recognition, which is then added to a larger system. According to subject 3, this is like purchasing a completed service. During the empirical gathering, it could be concluded that all organizations worked with development projects, whilst Evry also worked with standardized AI solutions. Swedavia differed slightly compared to the other companies, because apart from developing AI solutions in house, they also purchased and implemented standardized AI solutions into their own organization. To clarify, Swedavia was the only company that expressed a combination of both purchasing and implementing standardized AI solutions in addition to their in-house development.

4.2.2 System Development Methodologies

Purchasing and implementing a standardized AI systems is relatively straight forward. Subject 3 described this process as one where the customers express a need for a certain function and Evry realizes that it can be fulfilled via a standardized AI solution, consequently it is purchased and implemented. Because of this, standardized AI solutions will not be covered further in this chapter. The focus will instead lay on the development of AI solutions, which is something that all of the participated organizations work with.

4.2.2.1 Development Approaches

A common denominator was that all organizations worked within the iterative and agile framework. This was motivated by the difficulties to promise results, behavior and timeframe when working with AI solutions. On the aspect of difficulties with promising results, subject 3 at Evry explains:

> A major part of machine learning projects is to find if there are any connections and correlations in the data, and since it is impossible to know this beforehand, it is impossible to promise a connection before it is found…. So if one would use, for example, the waterfall model in a situation where the result is decided beforehand, it would not work, because how can you promise a result before you know if there are any connections in the data?

This is supplemented by subject 5 and subject 6 at Lexplore, who motivates the need for iterative development by explaining the difficulties of predicting behavior:

> It is very important to work iteratively, because you continuously need to achieve control over how the system is behaving… In particular when new data is inserted, how do you make sure that it upholds the same standard? It is a difficult problem to solve….

Centiro also expressed the difficulties of predicting results and behavior:

> I would say AI solutions have a higher percentage of failure compared to other software development projects… You know, if someone tells me to explain what this model can do in two or three weeks, I would tell them that I do not know… I need to test the idea, and see it visually, and then continue…

This is why iterations are so frequently used, because it allows the projects to progress and change direction based on what has been discovered. This is also what makes the development process agile, because the sense of direction can change after each iteration. According to Swedavia, an iterative development process and requirement analysis is a fundamental requirement to even be able to develop AI solutions. The need for an iterative and agile framework is hence motivated by difficulties that are particularly unique to AI solutions development, such as the difficulties with promising results and behavior as well as estimating timeframes. None of the organizations stated that they had to change their overlaying systems development methodologies as a result of the AI development projects. In other words, they were already accustomed to development in the iterative and agile framework. Subject 4 at Findwise states:

> We have not experienced the need for a unique development method for the AI projects, they have been managed in the same way as more traditional development projects.

However, there were certain methods within the agile framework that had been prioritized differently by the organizations. Evry, Findwise and Swedavia stated that Scrum, or a sort of self-made variant of it (that was a combination of several agile methods), was used during the development project. Centiro, on the other hand, had embraced Kanban. According to Centiro, Kanban is more suitable for AI development than other agile methodologies. Subject 1 motivates this by stating that it is because it is extremely hard to estimate times during AI development.

4.2.2.2 Teams

Moreover, all organizations managed the AI development in teams. The composition of competences within a certain team differed between the organizations. Subject 1 at Centiro explains that they work on several AI projects at the same time, usually together with other teams at Centiro. The same approach was used by Findwise, where a team consisting of two- to five experienced and competent (in AI) employees was responsible for the projects. At Evry, the teams consists of a mixture of competences, but they have employees (data scientists) at the company that work specifically with AI.

4.2.3 Requirement Analysis Process

4.2.3.1 Requirements Analysis

The general assumption by all involved organizations is that the process of requirement analysis does not differ between development of AI solutions compared to traditional solutions. Requirements analysis is regarded as an important aspect of the development projects, and although it is always included, the extent to which it is incorporated in a development project varies based on the project at hand. Subject 2 at Centiro explains that from their perspective, this is a result of the varied nature of most development projects, hence the level of requirements analysis is decided based on situational aspects. Primarily, it is the aspect of novelty in the development project ahed that decides how extensive the requirements analysis needs to be. Subject 2 at Centiro explains:

> In the situations where requirements can be expressed and documented efficiently, less requirements analysis is needed, compared to projects where requirements can not be defined up-front. In those cases, we need to work closely with the customer with design thinking, workshops and prototypes in order to find the right solution.

This is further discussed by subject 4 at Findwise, who explains that beyond novelty, the size of a project is also a determining factor: ”The level of detail to requirements analysis is typically greater in larger projects that are technically heavier, compared to smaller projects”.

4.2.3.2 Requirement Management Methods

Despite that requirements analysis is an inclusive phase during development, no organization used a purely specific requirements management method. Instead, requirements were typically expressed in the iterations via different frameworks, such as Kanban and Scrum. Every participant declared that an iterative and agile approach to the requirement analysis is absolutely necessary when developing an AI solution. In other words, requirements are expressed over the course of iterations, and are prioritized based on their relevance. This approach to requirements is prevailing among all the objects of study and no-one expressed that a specific method is used. Centiro, Evry and Findwise states that the requirements analysis phase differ from team to team, and that it can change over time within a specific team. The teams in those organizations are free to take parts from certain requirements management methods and mix them how ever they like.

It could also be understood that the knowledge of specific requirements terminology were mixed. Functional and non-functional requirements were not commonly known among the organisations, with the exception of one company that specifically expressed that they incorporated those terms in their daily operations. The same goes for sensational requirements, of which one organization stated that the term was well known but seldom used in practice.

Subject 1 at Centiro explains that their AI team seldom have direct communication about requirements with the end-users or customers. They explain that since the AI solution is often only a small piece of a much larger system, the teams that handle the traditional parts are often responsible for the requirement communication with the end-users and customers. To clarify, subject 1 believes that the requirements that are expressed by the end-users and customers are received by the traditional development teams, and related requirements to AI are later transferred on to the AI team. Swedavia discussed that they put together a sort of management team when receiving a request of a new system or a new function. This team communicates with the end-user to fully understand what they require from the new solution. This applies both to traditional projects and to AI projects. Furthermore, they added that the requirements analysis for AI solutions requires denser reconciliations, in comparison to traditional development.

Generally speaking, the companies explained that they have no specific method for prioritizing and structuring requirements. When asked about if they use well-established methods or techniques for prioritizing such as the MoSCoW technique - they all answered a clear no. The companies stated that they do naturally prioritize and structure the requirements, but without any specific method or technique. Instead requirements are usually expressed in different forms such as in Backlogs (for Scrum), Models, Roadmaps and through management softwares such as Trello and Jira. Through these kinds of visualizations, the customer or end-user can be a part of the development process and give new suggestions or requirements between iterations in the agile and iterative work environment. For example, Findwise describes how they use the software Trello for their requirements process and for keeping in touch with the customer. Subject 4 explains:

> Customers can log into Trello just like us… It is a web-based tool so often we sit in our status- or reconciliation meetings and go through the status of Trello

> and then say: "in this sprint we were supposed to finish these ten requirements”, and hopefully we have completed all of the requirements at the end of the sprint… And the customer can log in by themselves in the system [Trello] and actually see at any time, ”okay, Findwise is working on these five requirements right now, three are parked and the rest are waiting to be retrieved from the backlog”.

Subject 4 continues to explain that they focus on using backlogs as a part of their version of the Scrum framework. This is done together with the customer.

> Requirements that are present in the backlog have been established on a high level, these are then divided into sprints, and before each sprint we delve down to details of each individual requirements.

To give a third example of how the organizations work, but from the perspective of a start-up, where their AI provides almost all of their business value, Lexplore states:

> We have a product owner at the company that reviews comments from customers and from sellers - what to improve and change. We then try to create a roadmap that controls resources and what we prioritize… Prioritizing everything is absolutely the most difficult aspect, because the requirements are constantly changing… For this reason, it is one of the most arduous challenges we have as a start-up business.

4.2.3.3 The Importance of Data

A major difference in requirements that was subsequently understood from the interviews was the fact that the developers need to possess a large quantity of quality data for developing an AI solution. A traditional system can be told what to do and how to act in certain situations purely based on code, but an AI system needs to be taught how to act in new situations. Therefore, if the company do not possess a sufficient amount of data, it is impossible to develop the AI solution to begin with. Subject 1 at Centiro concludes that the number one requirement, or even prerequisite, for AI solutions is the quality and amount of data. Subject 1 explains this via a parable of teaching a baby what things are, for example that a banana is in fact a banana, and not something else. Subject 1 at Centiro explains:

> You need to tell the baby that this is a banana, so he can recognize bananas. The same thing goes with machines. But unfortunately, it's not enough to give them [machines] one banana. You have to give them multiple bananas! You have to have multiple pictures of bananas for the machines to understand what a banana is.

In other words, it is of great importance that the developers possess a large set of quality data that can be used to teach the AI. Subject 1 continues:

> The thing with AI is that those [AI] systems or solutions learn from the data itself. So you might have a very good idea, but if you do not have the proper data to train or to teach the machine, then you wont be able to do it. So that is why data is the number one requirement.

These statements are reinforced by subject 7 and subject 8 at Swedavia, who expressed that in order to develop an AI solution that can yield a highly precise result it is necessary to have access to a large set of data. They explained this by providing an example of an AI solution that will predict the waiting time at the airports that they operate. The AI solution is already in use today, but Swedavia expressed that they wanted to improve its functions further, and that the way to do that is to collect more history (i.e, data).

4.2.3.4 When is an AI solution completed?

It can be of great difficulty to determine when an AI project is finished. As previously mentioned, our interviewees states that it is impossible to promise results, behavior and timeframe during AI development projects. This is motivated by the difficulty to predict how- and if the model will work. According to the organizations, this obscurity is prevalent through the whole development process. In other words, even when a model has been developed and is regarded as relatively complete, it will almost always inhabit certain flaws. Lexplore explains:

> Our model is not 100% accurate, because it is almost impossible to achieve that level of accuracy… so in the few instances when it [the model] does something wrong, it often does something very wrong… so it can be like 98-99% accurate, and that's fine, until the error occurs, because that undermines the confidence for the model.

In other words, Lexplore explains that it is almost impossible to achieve an AI solution that works with 100% accuracy. This view is universally shared amongst the organizations. According to Evry, the level of required accuracy is decided based on what problem the AI will solve. Evry explains:

> If you are faced with the choice of amputating a leg, and it [the AI solution] tells you to do so with 98% accuracy, should you then do it? I mean, it [accuracy] is very dependent on the project at hand… in some cases 80% might be enough…

In other words, the level of required accuracy varies between different projects. Findwise provided another example of this:

> For example, we have developed a translation service for a customer… even if the translation service is not 100% yet, it can be regarded as good enough if we see that the translation quality hovers around 70-80%… it is the same for, for example, Google translate…. so it is possible to come a long way, but not 100% all the way, so there are still problems with the quality [of AI solutions].

This is what further complicates the aspect of expressing requirements regarding results, behavior and timeframe. Evry explains that during a traditional systems development project, the developers can clearly see and say to the customer that a certain percentage of the software is completed (for example that 10% of the software has been built), but since AI is most commonly used for finding correlations in data, it is almost impossible to tell how much of the AI solution that is completed at a given time.

4.2.3.5 Users

For the development of AI solutions, users were regarded as relatively incognizant by all of the organizations. A common theme is that the end users were typically regarded to not possess sufficient technical knowledge of the possibilities and limitations of AI technologies. As a consequence, the need for AI development is realized by the organizations rather than by the end users. Hence, the end users seldom express that they specifically want or need an AI solution. Subject 1 at Centiro explains:

> Sometimes they [users] do not have the knowledge about how the solution will be, right? So what you need is to first understand their problem, and you need to understand their exact objective, because when you are building an AI solution you need to solve their exact problem… So first we look at that, and then we have to answer if we have the proper data to solve that automation problem.

This is further confirmed by subject 4 at Findwise, who states:

> Very few of our customers actually possess the knowledge or experience to understand AI and to see its possibilities… This is a consequence of that the technology is so new, or obscure, as you [Anton Olsson & Gustaf Joelsson] put it…. Therefore, a large part of our job is to find and create the technical solution [the AI solution]…

According to Findwise and Centiro, this requires the AI development team to possess a greater understanding of the core business as well as business value, in comparison to other development teams. There are of course exceptions, which subject 3 at Evry pointed out. According to subject 3, there are mainly two types of users:

> I think there are two main types of users, those who have realized that machine learning solutions can help them reach their target, and those who say ”help us to find a target”.

In other words, there are those who understand that an AI solution would be a beneficial solution for their specific problem. However, subject 3 explains that the problem with this is that even though they have realized that an AI solution can be used, they are still unable to express specific requirements. This has to do with the obscure nature of the AI. In other words, that it immensely difficult to promise results, behavior and timeframe, which was covered in the previous chapter.

Furthermore, in regards to those who do not posses such knowledge, the common conception is that there is no need to explain exactly how the AI solutions will work. For example, subject 1 at Centiro states that they do not put a lot of focus on trying to explain how the AI will function for the end users. Subject 1 explains:

> When was the last time you did a google search? And when you did, did you understand how the AI behind the google search worked? My point is, I do not think users have to understand how the AI solution works - but what they should understand are the consequences of it [the AI solution].

This is further backed by subject 6 at Lexplore, who explains:

> I do not think the users need to know the details of the AI solution…. They need to understand and be able to analyze the results presented by the AI, but they do not have to understand how they [the results] have been generated.

Swedavia added: ”They [the users] need to understand up to a certain level until they have confidence in the produced output”.

To clarify, they state that it is not important to covey how the AI solution will function, but instead which consequences it will have, what problems it will solve and what results it will provide. For this reason, subject 1 concludes that the AI team at Centiro does not work particularly with requirements analysis from a user perspective, apart from gaining insights to which problem they need to solve, or in other words a sense of general direction for the development project. The same goes for the other organizations as well. Subject 4 at Findwise explains: ”…I would say that if I compare our AI projects to our traditional development projects, the users are less involved. This has to do with the difficulties of the technology”.

4.2.3.6 User Participation During Development

One organization stated that they began incorporating the end users more as the project had progressed enough to exhibit a relatively functioning visual model. According to the organization, this aided the development project. Lexplore explains:

> During the development process, we initially only performed tests together with the kids, not the end users [teachers and specials needs pedagogues]… so the end users were never particularly involved… since they [end users] had no particular connection to the tests, they did not quite understand the results themselves… These days, we let the teachers [end users] perform the test as well, which provides them with a whole new level of understand for both the results and what they actually mean. So there was a massive difference in the understanding of the AI solution when they [end users] had participated in the tests themselves.

After the statement above was expressed, they were asked if that enabled the end users to express better requirements that were useful for the development process, to which Lexplore answered: ”Absolutely!”.

5. Discussion

In this fifth chapter, the empirics is related to the theoretical framework. The discussion is continuous and categorized based on different overlaying observations. The discussion is also performed from the perspective of the research question and purpose of this work.

5.1 Requirements Analysis

The empirical gathering suggests an interesting phenomena regarding requirements analysis during development of AI solutions. The theory suggests that requirements analysis is vital for any systems development project, and a lot of focus is directed towards the users. The theory continuously highlights the importance of creating possibilities for the users to understand how the new system should function, as well as what it should do, because without such knowledge it is impossible to determine sufficient technical requirements. The problem with this has been that even traditional systems are very difficult for users to fully understand, which leads one to believe that it is immensely difficult in the case of obscure AI solutions. Based on the theory, one would believe that the AI solutions would have to be technically explained for the users in order for the development of such projects to be successful. I.e, that a deep understanding of the AI solution itself is necessary.

However, the empirical data suggest something else. It was universally agreed that AI solutions are merely a part of a larger system, usually with the goal of automizing a certain function. As Centiro explained, the terms AI system and AI solution are interchangeable, because an AI system is a system that utilizes an AI function (or; solution). This prompts one to realize that the AI solution in itself must not be understood by the users, since user requirements are primarily directed towards the larger system. Instead, the users must understand the consequences of the AI and the function it contributes with to the larger system. Consequently, requirements analysis with regards to development of AI solutions can be argued to be more concerned with technical requirements rather than user requirements. This does not infer that users have no part in requirements analysis during AI development, because they still do. Rather, it arguably shifts the focus from the users (which is where the focus often is situated during traditional systems development according to the theory) to the technological aspects.

5.2 The Fundamental Requirement

To create and develop an AI solution that fulfill the requirements made by Russell and Norvig (2010), (to act similar to an intellectual being), the theory suggest two different approaches - supervised learning and unsupervised learning. Ebert and Louridas (2016) states that supervised learning can easily be described as a method where you give a machine a large set of data with both ”questions” and ”answers” and the AI will, based on this, learn how to act. When looking into the empirical data gathered from the interviews, it shows that this method is by far the most commonly used one, but it is also supplemented with unsupervised methods. Subject 1 at Centiro and subject 5 at Lexplore clearly points out that it is of great importance to posses and use large sets of quality data when developing an AI using the supervised learning method. Centiro states an example that for a machine to learn and recognize an item, the training set of data which is used for teaching the AI must consist of a vast quantity of quality pictures of the item which can result in an AI that can recognize the item in question from new pictures. Subject 1 also confirms the theory by saying that it is, by any mean, impossible to develop an AI system if the developers do not possess a large amount of useful data. The interviewees at Lexplore fortify Centiro's answer by explaining how their company improved drastically when more higher quality data was gathered.

Data is the single most basic requirement that is needed when developing an AI. In the theory, chapter 2.6.1.2, Ericsson (2008) talks about four different subcategories of non-functional requirements of which one is reliability. This specific subcategory is mainly focused on that a system should be reliable with few errors and as little downtime as possible. Reliability can be closely connected to how data is the most fundamental requirement of them all. The theory addresses as an example that an error or miscalculation should not occur more than a specific instances each year. This is directly comparable to the companies need for large quantities of quality data when developing and teaching an AI. According to the commonly used supervised learning method, if an AI have only been able to get access to a low number of pictures from the previous example, it will perform significantly worse than if it would have been trained with a larger training set of data with a high number of pictures. With other words, a large set of quality data would result in that the AI could identify an object an object in a new picture than if a small set of data would have been used.

5.3 Standardized AI Solutions and AI Development Projects

The theory suggests that there are two main approaches to information systems; standardized- and uniquely developed (also; proprietary, as mentioned in the theory) information systems. As described in the theory, a standardized system is developed to satisfy and comply with requirements from several companies within a specific industry, while a uniquely developed information system is tailored to an individual organizations specific needs. During the empirical collection, it was understood that the same principle can be applied to AI solutions as well. In other words, there are many standardized AI solutions available on the market, for example image recognition, chatbot functions and translation services. Evry pointed out that the majority of their projects concerning AI solutions are projects where a standardized AI solution is implemented. Evry motivated this by emphasizing the vast amount of resources needed in order to develop an AI solution from scratch.

Hence, it was understood that requirements analysis varied slightly depending on if it was a standardized AI solution or a pure AI development project. Particularly since standardized AI solutions are focused on incorporation and implementation, while pure AI development projects are a lot more concerned with what the name infers; development. As already established, AI solutions are often smaller parts (or functions) of a larger system. Because of this, Evry described that the implementation of a standardized AI solution as relatively straight forward. I.e, it is merely a process of implementing a small function in a much larger system. Often, their customer expressed requirements for certain functions that Evry realized could be solved using a standardized AI solution, and consequently it was incorporated into the final system. This process varies a lot in comparison to the AI development projects, particularly since our interviewees expressed major difficulties with promising fundamental results, behaviors and timeframes during those projects. Thus, one can draw the conclusion that standardized AI projects share much more in common with traditional projects in comparison with pure AI solution development projects.

5.4 Users and Explainable AI

According to the theory, AI solutions are a very difficult subject for users to grasp. This is further backed by the empirics collected in this work. In general, the participating organizations (specifically: AI development teams within the organizations) stated that they do not consider user input as something essential during the development of AI solutions. This is not to say that users are not involved during the iterations. A general sense of direction is required from the customers and/or end users, but the exact direction of the development was decided by the AI development teams. This is motivated by the common conception that users are limited by their understanding of the technological possibilities and limitations of AI, and therefore they are unable to express sufficient requirements. It is further fueled by the obscure nature of AI, which means that not even the developers are able to know if- and how the AI solution will work beforehand. As a consequence, the participants stated that they are not concerned with explaining how the AI solution will function to the customers and/or end users, since there is no need for them to understand this. Instead, it is up to the AI-teams to realize what the customers and/or end users actually want, and to consequently develop an AI solution that can solve their exact problem. In short, the focus with regards to the users are that they should understand the consequences and results of the AI, not the core technology behind it.

However, what is unique is that understanding an AI solution (and the conclusions and decisions it makes) is not an easy task for the customers and/or end users. As covered in the theory, this is why explainable AI (XAI) has gathered a lot of traction in recent years. This is particularly understandable since (as previously established) the customers and/or end users almost always comes in contact with the consequences of the AI. Despite this, XAI was not directly adopted amongst the organizations. There were no conclusive motivation for this. Centiro explained that XAI is a common phenomena for AI used in, for example, the financial market while it is less frequent in other markets. However, this does not mean that the organizations develop their AI solutions in an unexplainable fashion, it simply means that they do not follow the exact XAI criteria.

An interesting phenomena that was observed was that even though the end users were not considered to be a part of the development process, some organizations stated that they had experienced a vast improvement of the development process in instances where users were more engaged than usual. The most striking example of this was Lexplore. They did not initially involve the end users as they were developing their model, and focused instead on performing tests with other subjects. This ment that when results from the AI system was presented to the end user, they did not understand the results properly. However, as Lexplore started to involve the end users (for example, via letting them perform the tests themselves), they experienced a vast increase of the end users understanding of the results and the motivation behind them. As a consequence, the end users were able to express better requirements that aided the AI development process positively.

5.5 Requirements Terminology

Furthermore, the theory suggests that there are many different types of requirements for analysts and developers to keep in mind as they preform the requirements analysis. Particularly the ones mentioned by Eriksson (2008), which are functional, non-functional as well as normal, expected and sensational requirements. Eriksson (2008) argues that it is necessary to fulfill as many of these requirements as possible in order to perform a successful development project. This is further backed by theory emphasizing the importance of requirements gathering.

However, the empirical collection entails something interesting. Many of the participating organizations were not familiar with the particular terminology of the different types of requirements. Perhaps this can be blamed on the lack of a theoretically pure requirements management method amongst the organizations, or that the terminology was momentarily forgotten as the interviews were conducted. However, one organization did state that they worked frequently with the different types of requirements, and that the extent to which they used them differed between projects. They also added that even though the terminology was well known, some requirements - particularly sensational requirements - were not commonly used in practice, even though the knowledge about it still existed.

Yet, one could argue that despite the general lack of knowledge about the terminology, the organizations still fulfill the requirements. For example, the definition of sensational requirements mentioned by Eriksson (2008) are requirements that are not expressed nor' expected, if they are met it does not lead to dissatisfaction, while if they are met it leads to extreme satisfaction. This definition per se is arguably almost a definition of the main goal of AI solutions in itself. In other words, the theory and empirical gathering suggests that users are typically not able to express what they want or expect from an AI solution. Instead, users mainly deal with the consequences (i.e, the automation of previously manual labor tasks, i.e, elevated business value) from the AI solution. Thus, a successful AI solution in itself arguably becomes a fulfillment of sensational requirements.

5.6 Common Ways of Dealing with Requirements

In the empirics, it was established that an iterative and agile approach was used by all organizations. Since no-one stated that they exhibited a pure requirements management method, questions arose regarding how requirements were structured, prioritized and managed (which is an important aspect according to the theory). The universal answer was that requirements were managed as a part of the overlaying systems development framework. Although all organizations used the iterative and agile methodologies, the approaches within them were slightly different. The most common approach was Scrum, or a Scrum hybrid that incorporated a variety of features from other agile frameworks. Another organization motivated that Kanban was a more suitable framework for AI development in comparison to Scrum. Consequently, they used a Kanban hybrid. Lastly, two online management softwares presented in the empirics were used to manage requirements. As the theory describes, all of these frameworks are characterized by their adaptive and alterable nature.

This prompts one to realize that the obvious usage of agile frameworks points towards the need for a dynamic working environment when conducting AI development. The need for a dynamic framework is, once again, a consequence of the obscure nature of AI development projects, and the difficulties they inhabit with promising results.

6. Conclusion

In this final chapter, the research question is answered and suggestions of future research is presented.

6.1 How is the requirements analysis phase executed during the development of AI solutions?

The results of this study suggests that the requirements analysis phase is executed the same way as during development of traditional solutions. I.e, that the requirements analysis phase does not differ in the early stages of the development project during the development of AI solutions in comparison to traditional solutions. In particular, this is contributed by the fact that AI solutions often serves as a small part of a much larger system. None of the organizations stated that they had to change their overlaying systems development methodology as a consequence of developing AI solutions. Since no organization inhabited a specific requirements management method and instead managed requirements through their particular systems development method, this infers that the requirements analysis does not differ either.

However, as lifted in the discussion, we have been able to find various differences that we deem to be particularly unique to requirements analysis when developing AI solutions. These can be categorized into (1) the need for an iterative and agile systems development process, with an associated iterative and agile requirements analysis, (2) the importance of having a large set of quality data, (3) the relative deprioritization of user involvement, and (4) the difficulty of establishing timeframe, results/feasibility and the behavior of the AI solution beforehand.

6.1.1 Systems Development Methodologies

The universal systems development approach used by the objects of study is primarily based on the agile and iterative methods. As described in the discussion, an AI solution is commonly developed using the agile and iterative frameworks such as Scrum and Kanban, but with influence from other frameworks. This finding leads us to conclude that these methods and frameworks also affects the requirement process in whole. The interviewed organizations base their entire requirement management on the described methods, this is motivated by the need to be receptive and adaptive to change. Through the use of Scrum, Kanban or a mix of similar development frameworks, the companies are able to comply with the described changes specifically required in AI development.

A common theme amongst the majority of the involved organizations is the creation and usage of specific AI teams in each development project. Although the composition of the AI teams varied, they can be described as either purely AI concentrated or a mix between different competences depending on the company's business approach.

To conclude, the more abstract nature of AI solutions leads to an even more iterative requirement analysis in comparison to traditional development. An iterative development process and requirement analysis can be regarded as a fundamental requirement in order to be able to develop AI solutions.

6.1.2 Data

Compared to traditional development, there is a much greater emphasis on having a vast amount of quality data when developing AI solutions. Data is the number one requirement, or even prerequisite, for a successful AI solution development project and is profoundly necessary for successfully teaching an AI solution how to act. One organization confirmed that their decision to acquire new and a larger amount of quality has resulted in a higher quality product (I.e, a better working AI solution).

6.1.3 Users

Users are generally less involved during AI solution development projects compared to traditional development projects. This is a result of various aspects, but the most predominant one is that users are typically regarded as relatively incognizant when it comes to understanding AI technologies and its limitations and possibilities. The majority of the objects of study express that it is not necessary for the customer and/or end-user to understand how an AI solution functions, but instead it is to focus on the consequences the AI solution. However, it should be stated that one organization saw a great improvement of their AI development after they incorporated the end-users more in their development process. Despite this, it can be concluded that requirements analysis is commonly more technology focused rather than user focused during the development of AI solutions.

6.1.4 Results, Behavior and Timeframe

When performing AI solution development projects, it is almost impossible to determine results, behavior and timeframe beforehand. This has to do with the obscure nature of the AI technology. For this reason, end users are not able to express requirements regarding such aspects. This differs a lot in comparison to traditional development, where a user for example might express requirements for certain results or behaviors. To clarify, it can be concluded that during AI solutions development projects, it is nearly impossible to instantly provide a timeframe or make promises regarding how the solution will behave - or even if it possible to develop an AI solution at all.

6.2 Suggestions for Future Research

During our empirical gathering, we were presented with the understanding that there were many different types of AI solutions available on the market. The approach used for this study has been to speak of AI solutions in general terms. However, it would be interesting to research how different specific types of AI solutions differ from one another regarding requirements analysis. For example, to delve deeper in the differences between standardized AI solutions and pure AI solution development projects. Furthermore, it would be interesting to investigate how corporations will manage personal data in the future. Our study clearly shows that a large amount of quality data is crucial in order for an AI to be developed, which leads one to realize that data is becoming an increasingly more valuable resource. Will this affect how organizations store and use data? Will it incentive organizations to pass legal boundaries in order for them to achieve more competitive AI solutions?

Finally, it should be added that this study was limited by time and resources, and that a more profound study investigating the same field would be of great value and interest.

References

Alshenqeeti, H. (2014). Interviewing as a Data Collection Method: A Critical Review. *Journal of English Linguistics Research,* 3(1), 47. doi:10.5430/elr.v3n1p39 [2018-11-01]

Anderson, D. & Carmichael, A. (2016). *Essential Kanban Condensed*. 1st Edition. Seattle, Washington: Lean Kanban University Press.

Balaji, S & Sundararajan Murugaiyan, D. (2012). Waterfall vs V-model vs Agile: A comparative study on SDLC. *International Journal of Information Technology and Business Management,* 2(1), pp. 26-30. https://www.jitbm.com/Volume2No1/waterfall.pdf [2018-11-15]

Beck, K., Beedle, M., Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R., Mellor, S., Schwaber, K., Sutherland, J. & Thomas, D. (2001). *Manifesto for Agile Software Development*. Agilemanifesto. http://www.agilemanifesto.org/ [2018-12-15]

Bryman, A. & Bell, E. (2013). *Företagsekonomiska forskningsmetoder.* 2nd Edition. Malmö: Liber AB.

Bryman, A. (2018). *Samhällsvetenskapliga metoder.* 3rd Edition. Liber AB.

Centiro. (2019). *Centiro storsatsar på AI för smartare försörjningskedjor*. https://www.centiro.se/nyheter/2018/07/12/centiro-storsatsar-pa-ai-smartare-forsorjningskedjor [2019-01-09]

Chakraborty, A., Kanti Baowaly, M., Arefin, A. & Newaz Bahar, A. (2012). The Role of Requirement Engineering in Software Development Life Cycle. *Journal of Emerging Trends in Computing and Information Sciences*, 3(5), pp. 723-729. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.644.6803&rep=rep1&type=pdf [2018-11-27]

Cohen, L., Manion, L. & Morrison, K. (2000). *Research Methods in Education*. London: Routledge Falmer.

Copeland, B.J. (2018). Artificial intelligence. *Encyclopædia Britannica, inc*. https://www.britannica.com/technology/artificial-intelligence [2018-11-26]

Creswell, J. W. (2002). *Educational research: Planning, conducting, and evaluating quantitative.* 4th. Edition. Harlow: Pearson Education Limited.

Dastin, J. (2018). Amazon scraps secret AI recruiting tool that showed bias against women. *Reuters*, October 10th. https://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G [2018-11-07]

Dennis, A., Roth, R. & Wixom, B. (2012). *System Analysis and Design*. 5th Edition. Hoboken: John Wiley & Sons.

Ebert, C. & Louridas, P. (2016). Machine Learning. *IEEE Software*, 33(5), pp. 110-115. doi: 10.1109/MS.2016.114 [2018-11-20]

Eriksson, U. (2008). *Kravhantering för IT-system*. Lund: Studentlitteratur AB.

Evry. (2019). *About Evry*. https://www.evry.com/en/company/about-us2/about/ [2019-01-10]

Findwise. (2019). *Findwise - More than Enterprise Search*. https://findwise.com/en/findwise-more-than-enterprise-search [2019-01-09]

Ghosh, P. (2018). Machine Learning and Artificial Intelligence Trends in 2018. *DataVersity.* February 6th. http://www.dataversity.net/machine-learning-artificial-intelligence-trends-2018/ [2018-11-15]

Greengard, S. (2015). The Internet of Things. *Encyclopædia Britannica, inc*. https://www.britannica.com/topic/Internet-of-Things-The-2032782 [2019-01-25]

Hastie, S. & Wojewoda, S. (2015). Standish Group 2015 CHAOS Report - Q&A with Jennifer Lynch. *Infoq,* October 4th. https://www.infoq.com/articles/standish-chaos-2015 [2018-11-23]

Hedman, J., Nilsson, F. & Westelius, A. (2009). *Temperaturen på affärssystem i Sverige.* 1st Edition. Lund: Studentlitteratur AB.

Hoffer, J.A., George, J & Valachich, J. (2014). *Modern Systems Analysis and Design.* 7th Edition. Edinburgh: Pearson University Press.

Huisman, M & Iivari, J. (2006). Deployment of systems development methodologies: Perceptual congruence between IS managers and systems developers. *Information & Management,* 43(1), 29-49. doi:10.1016/j.im.2005.01.005 [2018-12-10]

Hurwitz, J. & Kirsch, D. (2018). *Machine learning for Dummies.* IBM Limited Edition. Hoboken: John Wiley & Sons.

Infosys. (2016). *Amplifying Human Potential: Towards Purposeful Artificial Intelligence* [white paper]. https://www.infosys.com/aimaturity/Documents/amplifying-human-potential-CIO-report.pdf

Jacobsen, D.I. (2017). *Hur genomför man undersökningar? - Introduktion till samhällsvetenskapliga metoder.* 2nd Edition. Lund: Studentlitteratur.

Jayaratna, N. (1994). *Understanding and evaluating methodologies : NIMSAD, a systemic framework.* London; New York: McGraw-Hill

Laudon K. C. & Laudon J. P. (2014). *Management Information Systems: Managing the Digital Firm*. 13th Edition. Harlow: Pearson Education Limited.

LeCun, Y., Bengio, Y. & Hinton, G. (2015). Deep learning. *Nature: International Journal of Science,* 521(7553), pp. 436-444. doi:10.1038/nature14539 [2018-12-05]

Lexplore. (2019). *Rapid Reading Assessment with Eye Tracking & Artificial Intelligence - Lexplore.* https://www.lexplore.com [2019-01-09]

Marr, B. (2018). 27 Incredible Examples Of AI And Machine Learning In Practice. *Forbes*, April 30th. https://www.forbes.com/sites/bernardmarr/2018/04/30/27-incredible-examples-of-ai-and-machine-learning-in-practice/#157ef0df7502 [2018-11-28]

Namrata, J. (2011). Software Development Life Cycle: A Detailed Study. *International Journal of Advanced Research in Computer Science,* 2(3), pp. 261-265. http://costello.pub.hb.se/login?url=https://search-proquest-com.lib.costello.pub.hb.se/docview/1443708677?accountid=9670 [2018-11-29]

Nieva, R. (2017). Facebook put cork in chatbots that created a secret language: Alice and Bob, the two bots, raise questions about the future of artificial intelligence. *Cnet*, July 31. https://www.cnet.com/news/what-happens-when-ai-bots-invent-their-own-language/ [2018-12-01]

Patel, R. & Davidson, B. (2003). *Forskningsmetodikens grunder: Att planera, genomföra och rapportera en undersökning*. 3rd Edition. Lund: Studentlitteratur.

Robson, C. & McCartan, K. (2016). *Real World Research*. 4th Edition. Chichester: Wiley.

Royce, W. W. (1970). Managing the Development of Large Software Systems. *Proceedings of IEEE WESCON,* 26, pp. 328-388.

Russell, S. & Norvig, P. (2010). *Artificial Intelligence: A Modern Approach.* 3rd Edition. Upper Saddle River: Pearson Education, Inc.

The SAGE encyclopedia of qualitative research methods Vol. 1. (2008). Comparative Research. Thousand Oaks: SAGE Publications. Available at: SAGE Publishing.

Schwaber, K. & Sutherland, J. (2017). *The Definitive Guide to Scrum: The Rules of the Game*. The Scrum Guide. https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf [2019-01-12]

Sheh, R. & Monteath, I. (2018). Defining Explainable AI for Requirement Analysis. *KI - Künstliche Intelligenz,* 32(4), pp. 261-266. doi:10.1007/s13218-018-0559-3 [2018-12-02]

Sommerville, I. (2011). *Software Engineering*. 9th Edition. Boston: Pearson Education, inc. Publishing as Addison-Wesley.

Stevens, W. P. (2002). Data Flow Analysis and Design. *Encyclopedia of Software Engineering*. doi:10.1002/0471028959.sof073 [2019-01-25]

Stone, P., Brooks, R., Brynjolfsson, E., Calo, R., Etzioni, O., Hager, G., Hirschberg, J., Kalyanakrishnan, S., Kamar, E., Kraus, S., Leyton-Brown, K., Parkes, D., Press, W., Saxenian, A., Shah, J., Tambe, M. and Teller, A. (2016). *Artificial Intelligence and Life in 2030: One Hundred Year Study on Artificial Intelligence, Report of the 2015-2016 Study Panel*. Stanford, CA: Stanford University. http://ai100.stanford.edu/2016-report [2018-11-19]

Swedavia (2019). *Role and mission*. https://www.swedavia.com/about-swedavia/role-and-mission/ [2019-01-27]

Appendix - Questionnaire (English)

General Information

1. Can you describe the company in general?
    1. What does the company work with?
    2. How many employees work on each development/implementation project?
    3. How do the project groups look like within the company during development/ implementation projects? (Number of participants, division of tasks, number of groups, etc.)
2. Can you tell a little about yourself and your role in the company?
    1. What is your working position and title?
    2. What is your education?
    3. What are your duties at work?
    4. What earlier work experience do you have?

AI

1. What kind of previous experience do you have of working with AI solutions?
2. Are there any employees who are specifically employed to work with AI solutions at the company?
3. Can you tell us about your current AI solution/s at the company?
    1. What is the purpose with the implemented/developed AI solution?
4. Why did the company choose to implement AI?
    1. Who saw the need and who made the decision?
5. Who are the users of the AI solution (s)?
6. Who participated in the acquisition/development of the AI solution (s)?
7. When did the company start working with AI?
8. How has AI affected your workplace and work routines after implementation?
    1. Does the AI behave as you intended?
    2. Any unforeseen problems?
    3. Do you work with Explainable AI?
9. What positive and negative aspects do you see that implementation of AI can have?

Requirements analysis for AI

1. Do you have any specific methods or standards for requirements analysis concerning "traditional" information system development and in that case, which ones?
    1. In general, do you perceive that requirements analysis methods are used?
2. Since AI solutions are relatively abstract (compared to traditional information systems), how do you think the requirement analysis phase is affected when an AI solution is to be implemented or developed?
    1. Do you have your own requirements analysis process specifically developed for AI solutions?
3. How does the requirement collection go when AI is to be implemented?
    1. How involved are the users?
    2. Do the users understand what the AI solution should do?
    3. How important is it for the users to understand what the AI solution should do?
4. How do you structure the requirements?
5. How do you prioritize the requirements?
6. How do you compile and document requirements?
7. How do you establish which requirements that should be included?

Users

1. In what ways are the prospective users involved in the requirement analysis phase?
2. How do you proceed to take into account the users' limited knowledge of AI? That is, do you make an abstract AI solution understandable to the users, so that they can express requirements?
3. An important part of requirement management is to (in addition to functional requirements) determine non-functional requirements (unspecified but desirable requirements) as well as sensational requirements (neither expected nor stated). How are these types of requirements dealt with when implementing AI solutions?
4. Do you work iteratively with requirements? If so, how do you ensure that users are involved with requirements through all iterations? For example, do you use prototypes, storyboards, scenarios or similar?

Final Question

1. To summarize - According to you, what are the biggest differences in the requirements analysis phase when an AI solution is to be implemented compared to a traditional solution?

Appendix - Questionnaire (Swedish)

Allmänt

1. Kan du beskriva företaget i allmänhet?
    1. Vad arbetar företaget med?
    2. Hur många anställda arbetar på varje utvecklings/implementeringsprojekt?
    3. Hur ser projektgrupperna ut inom företaget vid utveckling/ implementeringsprojekt? (Antal delaktiga, uppdelning av uppgifter, antal grupper, etc).
2. Berätta lite om dig själv och din roll på företaget
    1. Position
    2. Utbildning
    3. Arbetsuppgifter
    4. Tidigare arbetslivserfarenhet

AI

1. Vad har du personligen för tidigare erfarenhet av att arbeta med AI-lösningar?
2. Finns det personal som är specifikt anställd för att jobba med AI-lösningar på företaget?
3. Berätta om er/era AI-lösning/lösningar
    1. Vad har AI:n för syfte?
4. Varför valde företaget att implementera AI?

    A. Vilka såg behovet och vilka tog beslutet?
5. Vilka är användarna av AI-lösningen/arna?
6. Vilka deltog vid/i anskaffningen/anskaffningsprojektet av AI lösningen/arna?
7. När började företaget arbeta med AI?
8. Hur har AI påverkat er arbetsplats och arbetsrutiner efter implementering?
    1. Gör AI:n det som ni tänkt er?
    2. Några oförutsedda problem?
    3. Arbetar ni med Explainable AI?
9. Vad ser ni att implementering av AI kan ha för positiva och negativa aspekter?

Kravhantering för AI

1. Har ni några specifika metoder eller standarder för kravhantering för "traditionella" informationssystem och isåfall vilka?

    1. Generellt, uppfattar du att kravhanteringsmetoderna används?

2. Eftersom AI-system är relativt abstrakta (jämfört med traditionella informationssystem), hur anser du att kravhanteringen påverkas när en AI-lösning ska implementeras?

    A. Har ni en egen kravhanteringsprocess specifikt utvecklad för AI-lösningar?

3. Hur går insamlingen av krav till när AI ska implementeras?

    A. Hur involverade är användarna?

    B. Förstår användarna vad AI-lösningen ska göra?

    C. Hur viktigt är det att användarna förstår vad AI-lösningen ska göra?

4. Hur strukturerar ni kraven?
5. Hur prioriterar ni kraven?
6. Hur sammanställer ni och dokumenterar dessa krav?
7. Hur beslutas vilka krav som ska vara med?

Användare

1. På vilka sätt involveras blivande användares roll i kravhanteringen?
2. Hur går ni tillväga för att ta hänsyn till användarnas begränsade kunskaper om AI? Dvs. gör ni en abstrakt AI-lösning begriplig för användarna, så att de kan uttrycka krav?
3. En viktig del av kravhantering är att (utöver *funktionella krav*) fastställa *icke-funktionella krav* (ej uttalade men önskvärda krav) såväl som *sensationella krav* (varken förväntade eller uttalade). Hur behandlas dessa typer av krav vid implementering av AI-lösningar?
4. Jobbar ni iterativt med krav? Isåfall, hur ser ni till att användarna är delaktiga med krav genom alla iterationer? Ex. använder ni prototyper, storyboards, scenarios, etc.

Slutgiltigt

1. Sammanfattningsvis; enligt dig, vilka är de största skillnaderna i kravhanteringen när en AI-lösning ska implementeras jämfört med en traditionell lösning?

University of Borås is a modern university in the city center. We give education programs and courses in business administration and informatics, library and information science, fashion and textiles, behavioral sciences and teacher education, engineering and health sciences.

At the **Department of Information Technology,** we have focused on the students' future needs. Therefore, we have created programs in which employability is a key word. Subject integration, wholeness and contextualization are other important concepts. The department has a closeness, both between students and teachers as well as between industry and education.

Our **courses and programs** with a major in informatics are centered around basic concepts as system development and business development. In our wide range of specializations there is everything from programming advanced systems, analyze the needs and requirements of businesses, to conduct integrated IT and business development, with the common purpose of promoting good use of IT in enterprises and organizations.

The department is carrying out IT-related **research** within the university's research area called Business and IT. In terms of field, the research activities are mainly within **computer and systems science**. Particular areas of focus are **data science** and **information systems science**. Both scientifically and professionally-oriented research are performed, which among other things is manifested through that research is often conducted based on domain specific needs of business and government organizations at local, national and international arena. The professionally-oriented research is also often manifested through our participation in the Swedish Institute for Innovative Retailing (SIIR), which is a research center at the University with the aim of contributing to commerce and society with the development of innovative and sustainable trade.

UNIVERSITY
OF BORÅS

VISITING ADDRESS: JÄRNVÄGSGATAN 5 · POSTAL ADDRESS: ALLÉGATAN 1, SE-501 90 BORÅS

PHONE: + 46 33 435 40 00 · E-MAIL: INST.HIT@HB.SE · WEB: WWW.HB.SE/HIT