Aalto University
School of Business

Usability Testing Throughout The Application Lifecycle

Technology Management and Policy

Master's thesis

Joonas Kallankari

2013

Department of Information and Service Economy
Aalto University
School of Business

AALTO UNIVERSITY SCHOOL OF BUSINESS
Department of Information and Service Economy
Master's Thesis
Joonas Kallankari

ABSTRACT

Objectives of the Study

The aim of this master's thesis is to evaluate if and how usability testing could be used as a part of Application Lifecycle Management. This thesis addresses the current lack of evaluation and strategic management methods for Application Lifecycle Management, especially during the later phases of the application lifecycle. Scholars have acknowledged that usability testing has potential for mid-lifecycle use, but mid-lifecycle usability testing is not yet widely applied by organizations.

Methodology of the study

Prior research was used to build a framework for usability testing throughout the application lifecycle. The framework presents application lifecycle management in terms of the duration of the lifecycle as well as the different aspects of organization that manage, develop, and use the application. After developing the framework, a set of usability testing methodologies is suggested to be used with the framework.

The developed framework and the developed testing methodology were evaluated through applying the testing methodology in two separate company projects and collecting feedback from company representatives.

Findings and conclusions

Communicating with the users during the initial development and updating of applications is important and there is potential for usability testing as a part of Application Lifecycle Management. This research confirmed the notion that greatest utility from usability testing is gained at the initial development phase of the application lifecycle. The results suggest that usability metrics can be used to support decision making also during the later phases of the application lifecycle. The company interviews showed that usability metrics are suitable to some extent for strategic management of applications in form of evaluating vendors and applications in the long-term. However, in order to make usability testing more appealing as a supporting tool for Application Lifecycle Management, usability testing methodologies must be developed. Businesses have a need for objective usability testing results even with limited samples, as it is not easy for companies to find and engage real test users. Scholars must try to develop ways to introduce cost as a measure of severity of a given usability issue. This will add the value of usability testing outside of academia.

Keywords

Usability testing, Application Lifecycle Management, DeLone and McLean Information System Success Model, System Usability Scale

AALTO-YLIOPISTON KAUPPAKORKEAKOULU TIIVISTELMÄ
Tieto- ja palvelutalouden laitos 15.11.2013
Pro Gradu-tutkielma
Joonas Kallankari

ABSTRAKTI

Tutkimuksen tavoitteet

Tutkimuksen tavoitteena on arvioida käytettävyystutkimuksen soveltuvuutta sovellusten elinkaarten hallintaan. Tällä hetkellä sovellusten elinkaaren hallinnan tutkimuksessa ei ole laajasti käsitelty sovellusten arviointia tai strategista johtamista, ja tutkimus on keskittynyt sovellusten elinkaaren alkuun. Käytettävyyden asiantuntijat kertovat, että käytettävyystutkimuksella voidaan saavuttaa merkittävät hyödyt sovellusten elinkaarten keskellä, mutta organisaatiot eivät vielä laajasti käytä käytettävyystutkimusta elinkaarten keskellä.

Tutkimuksen metodologia

Aiempaan kirjallisuuteen perustuen ehdotan käytettävyystestauksen viitekehystä, jossa sovellusten elinkaarta käsitellään elinkaaren keston ja eri organisaation osien kautta. Viitekehyksen lisäksi tässä tutkimuksessa ehdotetaan käytettävyystutkimusmetodologiaa, jota voidaan soveltaa viitekehyksen raameissa.

Kehitettyä viitekehystä ja metodologiaa arvioitiin soveltamalla kehitettyä metodologiaa kahdessa erillisessä yritysprojektissa. Yritysten edustajat antoivat palautetta käytettävyystestauksesta yleensä, sekä osana sovellusten elinkaarten hallintaa.

Tulokset ja päätelmät

Loppukäyttäjien kanssa kommunikointi on tärkeää kun sovelluksiin tehdään muutoksia. Käytettävyystestaus sopii jossain määrin osaksi sovellusten elinkaarten hallintaa. Yritysten mukaan suurin hyöty käytettävyystestauksesta saadaan elinkaarten alussa, suunnittelu- ja kehitystasolla. Käytettävyysdatalla on kuitenkin käyttöä myös sovellusten strategisessa johtamisessa elinkaarten myöhemmissä vaiheissa. Käytettävyysdataa voidaan käyttää tukemaan sovellusten toimittajien ja projektiportfolioiden hallintaa. Jotta käytettävyystestauksesta tulisi yrityksille entistä mielekkäämpää, tulisi kehittää tapoja, joilla käytettävyysongelmien hinta tuodaan osaksi käytettävyysarvioita. Tämä lisäisi käytettävyystestauksen hyödyllisyyttä yrityksille.

Avainsanat

Käytettävyystestaus, Sovellusten Elinkaarten Hallinta, DeLone and McLean Information System Success Model, System Usability Scale

ACKNOWLEDGEMENTS

I was very lucky to have a wide support network of professionals to learn from and to discuss ideas with during this thesis project. This allowed me to really learn and develop during the year that this project ended up taking. I want to thank Esko Penttinen and the Real Time Economy Competence Center for facilitating and funding this research. I want to thank Sirpa Riihiaho, Johanna Kaipio, Tapio Haanperä and Jussi Rämänen, from the Strategic Usability Group in Aalto University, for their guidance into the world of usability, as well as for their support and feedback during the early phases of my research. I want to thank Timo Saarinen for supervising my thesis and giving insightful feedback about the theoretical foundation of my research. Finally, I want to thank representatives from both of the participated companies, as well as all of the test users who were a part of this research. My special thanks goes to the development manager of Company A, who gave his time to participate in the pilot test of the developed testing methodology, as well as for being very supportive during the project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

ALM	Application Lifecycle Management
APM	Application Portfolio Management
D&L IS SM	DeLone and McLean Information System Success Model
ICT	Information and Communications Technology
IS	Information System
PLM	Product Lifecycle Management
PM	Project Management
SUS	System Usability Scale
TAM	Technology Acceptance Model

1. INTRODUCTION

Information technology has become an integral part of business, no matter what the industry. We use information systems to plan and schedule; to communicate and manage; to produce and sell. Depending on the organization, one can find a variety of Information and Communications Technology (ICT) installations from heavy enterprise systems to small and agile specialized applications. Common to all of these information systems is that, sooner or later, they will get outdated. Companies and organizations are constantly facing pressures to engage in ICT development and evaluation of which applications to update and which to abandon altogether. The development (either the initial development or updating) of different applications can be done in-house, but is commonly outsourced, as are increasing number of the ICT operations. As complexity of ICT systems has increased over time, also managing ICT development has become challenging, as shown by the high number of ICT development projects that fail. Development is a big issue, but there is need for attention also after the initial development of an application. There is a need for a holistic management approach to ICT and the management of the portfolio of applications that companies need to juggle with. Application Lifecycle Management (ALM) addresses the challenges of managing applications over the long term. However, the term is often considered synonymous with cyclical development of software, and little focus has been put so far on the management of later phases of the application lifecycle.

Through this Master's Thesis, I try to suggest and evaluate usability testing as a tool to be used with ALM. There is a current problem with the very limited, and sometimes mistaken treatment of the concept of ALM. ALM is currently used to coordinate the activities and artefacts produced by different actors involved with ICT development and use. However, there is a lack of measurements that could be used throughout the application lifecycle. I consider prior research about possible Information System (IS) success measures in order to add a dimension of measurement to ALM. Analysis on literature suggests that usability testing could be a potential methodology to measure the different variables of IS success. In addition to measuring applications, usability testing can be used in designing IS interfaces to achieve desired IS success.

The aim of this paper is to evaluate if and how usability testing could be used as an ALM tool. Main contribution of this paper is to suggest a framework that combines ALM, IS success measures and usability testing. Through my research, I also propose a specific usability testing methodology that can be applied as part of ALM. Empirical part of my research evaluates the developed usability testing methodology through two separate company projects. In discussion part the initial model is updated based on company feedback. However, the empirical part of this research was limited and only covered a specific part of the application lifecycle with a relatively small sample. Further research is needed to evaluate the developed framework and usability testing methodology, especially in the governance aspect of ALM.

Research of information systems is a rapidly growing field. My treatment of information systems in this research stems from the business side of managing IS and is especially aimed to benefit those who are not specialized in technology. This is why terms such as Information System, Information Communications Technology and application are used quite liberally, even though I acknowledge that they are not the same thing. In this thesis I treat IS, ICT and application as terms for a technology based tool that the user uses in order to perform a certain task. Treating the terms synonymously allowed me to discuss and combine prior models about information technology. For my research, it is not significant whether or not the tool is connected to a network, is an independent entity, or an add-on to an existing system. The tool just needs to be separate enough, that it has a specific purpose and a separate lifecycle or existence from another system.

1.1. Research gap

The research gap my thesis aims to address is that of the limited treatment of Application Lifecycle Management after the software design phase. My first intuition when learning about ALM was that it is a higher level concept of planning and managing applications from the initial development to becoming obsolete, including technical, but also organizational aspects. However, treatment of ALM seems often to be that of using it synonymously with the software development lifecycle. This is especially evident outside of academic publications. The web dictionary Webopedia defines ALM as “the capability to integrate, coordinate and manage the

different phases of the software delivery process." (Webopedia, 2013). A different web dictionary, Technopedia, defines ALM as "the combined coordination of various development life cycle activities, such as requirements, modeling development, build and testing" (Technopedia, 2013). These are two examples of wrong definitions of ALM and illustrate that there are many who confuse ALM with development lifecycle management. This is perfectly understandable. The initial development is a very important part of ALM, and iterative design requires similar appreciation of continuously collecting, evaluating, and acting upon data that ALM does.

Scholars do recognize that ALM is more than just coordinating activities of the development lifecycle, but the academic literature is still heavily focused on the development phases of ALM. Kääriäinen (2011), in his dissertation that combines five publications about ALM, states how especially the operational and maintenance phases of application lifecycle have need for further suggestions.

Literature on ALM and earlier literature on Product Lifecycle Management (PLM) is also currently focused more on coordinating different activities of development and use, rather than measuring and evaluating the application (see for example Kääriäinen and Välimäki, 2011, and Rachuri et al., 2008). Common wisdom is that you cannot manage what you cannot measure. If ALM should become a useful management discipline, it also needs a dimension of measurement to it. Through my own experience, I believe that usability testing could address the current weaknesses in ALM.

Usability testing is traditionally seen as a design tool for product development. Even when usability testing is suggested to be used in an iterative manner with a lifecycle approach, the weight is in the design process (e.g. Rubin and Chisnell, 2008). Some progressive companies are already applying usability testing after the initial design phases of the application. Even then, usability testing tends to be limited to evaluating the design. The experts and scholars on usability are aware that mid-development (in an iterative manner during creation) and mid-lifecycle (in production) usability testing can have a strong positive effect on the finished product (Johnson et al., 2007). Conducting usability testing after the handing out from project

organization into production is still rare, even though continuous testing is recommended by scholars (Rubin and Chisnell, 2008).

Usability testing would be beneficial throughout the application lifecycle, not only during the design phase, but also after a product is launched to be used on the operational level. Aggregated usability testing data, collected with a uniform methodology over time, could also be used to support decision making in application- and vendor portfolio management. Applying usability testing as a management tool throughout the whole application lifecycle is an approach, which is not yet widely discussed.

1.2. Aim and scope of the research

The primary aim of my master's thesis is to answer to the lack of evaluation and strategic management tools for ALM. At the same time, usability testing's potential for in-production use is acknowledged by scholars, but not yet widely used by organizations. I will evaluate if and how usability testing could be used as a part of ALM. This requires a conceptual part, where I use prior research to suggest a framework of usability testing throughout the application lifecycle. The framework is then evaluated via empirical research.

The empirical part of my research is limited and only coveres a specific part of the application lifecycle with a relatively small sample. I applied the developed usability testing methodology during mid-lifecycle development projects. Emphasis of the empirical part is on the development aspect of ALM. Further research is needed to evaluate the suggested framework and usability testing methodology, especially in the governance -level of ALM. However, on a conceptual level, the suggested framework and conclusions apply to the whole of ALM. The developed ALM model can be applied to applications with internal and external users.

This aim and scope are articulated through the following research problem and research questions.

1.2.1. Research problem

There is a lack of measurement tools for managers overseeing Application Lifecycle Management. There is also a lack of tools for ALM for the mid- and late phases of the application lifecycle.

1.2.2. Research questions

1) What kind of usability testing methodology could be used to evaluate IS throughout the application lifecycle?

2) Will company representatives see the developed usability testing methodology as a useful tool for ALM at different phases and different aspects of the application lifecycle?

1.3. Research approach and structure of the thesis

This thesis has a conceptual and an empirical part and is qualitative in nature. The research consisted of developing conceptual framework for usability testing throughout the application lifecycle. After this, a specific usability testing methodology was developed to be used in the context of the suggested framework. The empirical part of the research was applying the usability testing methodology to two different ICT system development projects in separate companies and evaluating the tool via interviews from the company representatives. The actual usability evaluations (the reports to the companies) were left outside of the thesis because of confidentiality reasons. However, this does not affect the research, as the main research subject is not the companies' ICT systems, but rather usability testing's suitability for ALM.

Development of the usability testing methodology was done in co-operation with the Usability Laboratory of Aalto University School of Science. In addition to meetings with the members of the Usability Laboratory, I studied literature on usability testing concerning methods and limitations. The result is a testing methodology, that draws from prior research and experience of researches of the field. The methodology was tried out in a controlled pilot test with the project manager of Company A. After the pilot test, the methodology was modified based on feedback

from experts of the Usability Laboratory. The developed methodology was then applied to usability evaluation of Company A's new user interface for a sales channel of information. The same methodology was applied to Company B's ICT system development project of an in-house data entry and query system.

After using the developed testing method, I separately interviewed company representatives from both Company A and Company B in order to evaluate usability testing as a tool for ALM. Emphasis of the empirical part was to focus on the development aspect of ALM.

Figure 1 illustrates my research plan and the intended flow of the research. Different colors represents different sources of information. The structure of my thesis follows the same organization.

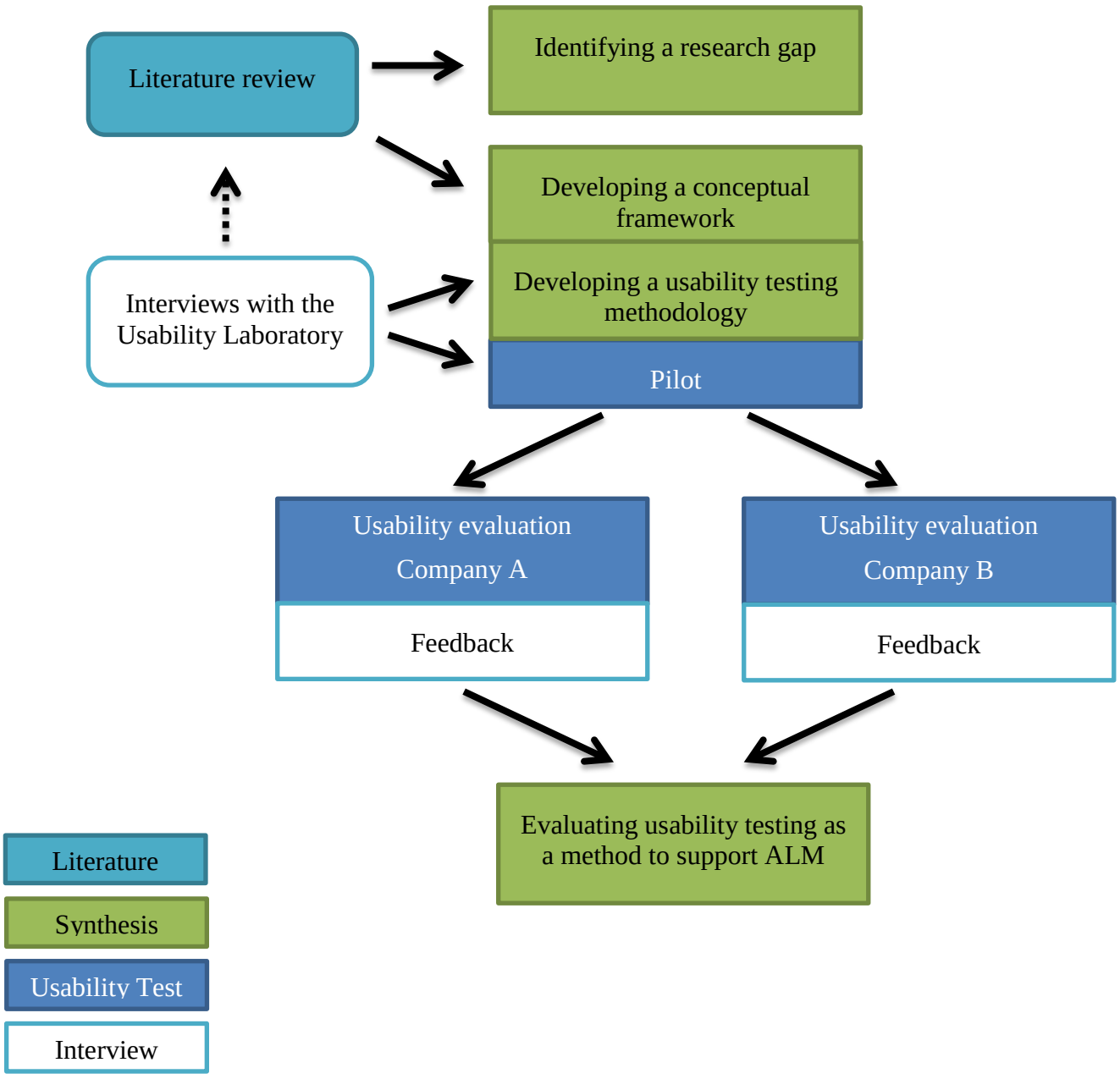


Figure 1: Methodologies and flow of the research

In the figure above, *Literature* refers to studying and reviewing literature. Prior research is presented in part **2.** (Literature Review).

Synthesis in the above picture refers to my own synthesis of information and development of own ideas. The green boxes illustrate the stages of identifying a research topic, developing the conceptual framework and the usability-testing methodology and evaluating them based on interviews with company representatives. It was important to identify and document a usability testing methodology that was applied to both companies. This allows for control as both companies are presented with the same concept of usability testing. Long-term use of usability testing for the governance level of ALM also requires a controlled methodology. Uniform measurement process allows for comparison of results of different usability evaluations over time.

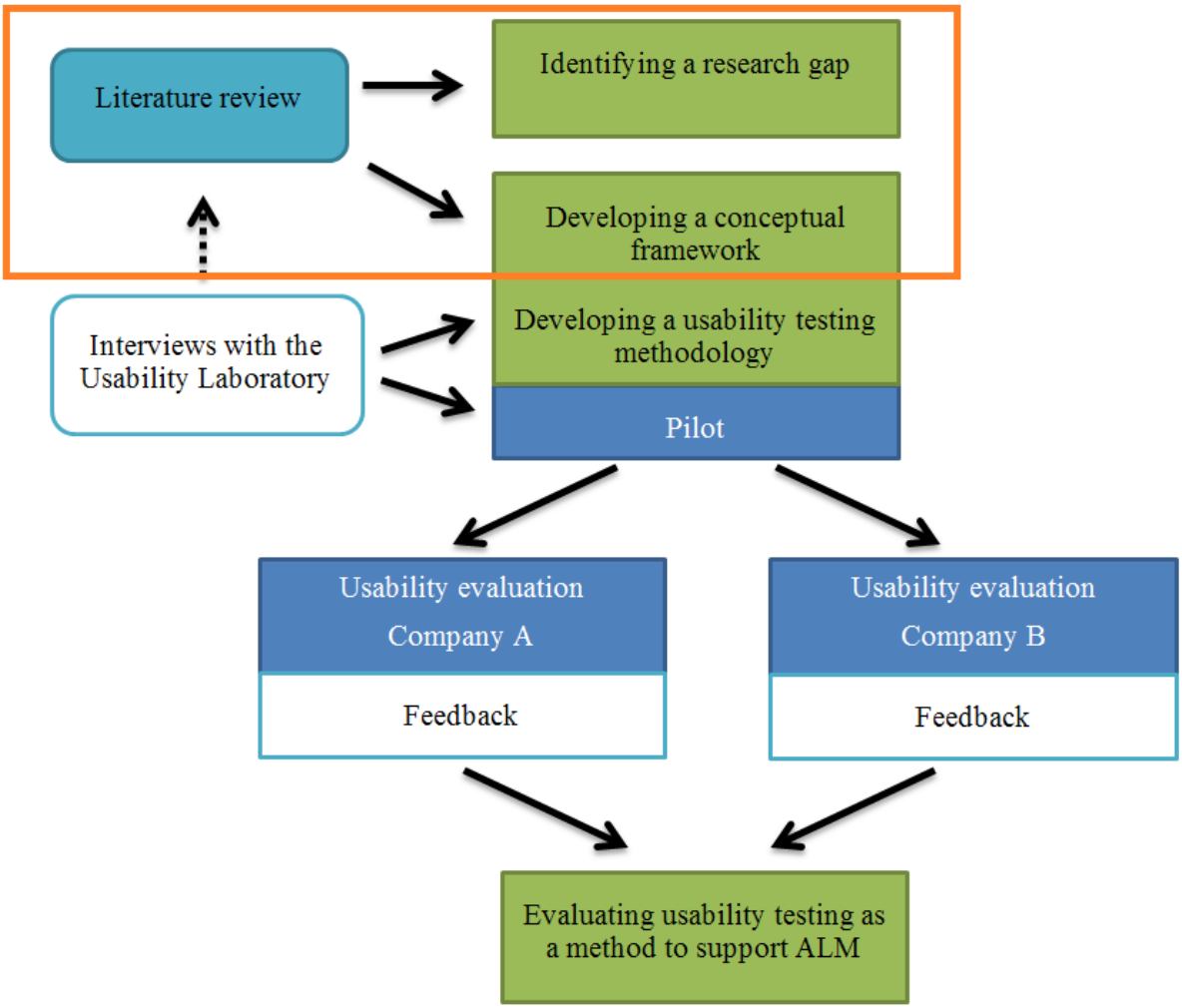
Major part of my research was to develop a usability testing methodology based on previous literature and interviews with business and usability experts. In the end, this methodology is a combination of existing methodologies. The methodology was used in same testing sessions to evaluate both old and the updated version of applications. The development of the usability testing methodology is described in part **3.1.** (Developing the usability testing methodology).

Usability Test in Figure 1 refers to the pilot and to the two actual usability evaluations conducted during my research. The results of the two usability evaluations are confidential and were left outside of this document.

The *Interview* steps in Figure 1 represent the interviews with the experts of Usability Laboratory as well as the project managers and other representatives from Company A and Company B. They were the actual research subjects of my master's thesis. I evaluated if usability evaluations is something that could be useful for real businesses in different phases of the application lifecycle. Secondary aim of the interviews was to get feedback for the developed usability testing methodology. It was important that the developed testing methodology was evaluated by the business representatives, in contrast to usability experts. Businesses have many more considerations to ALM than just having usable user interfaces. The results of the company interviews are presented in part **4.** (Results).

Finally, the results and research process is evaluated and summarized it in part, **5.** (Analysis and discussion of the results) and **6.** (Conclusions).

2. LITERATURE REVIEW



This literature review has two main components. Firstly, prior research is used to suggest a framework for usability testing throughout the application lifecycle. Secondly, prior research is used to help to develop a specific usability testing methodology to be used in the context of the framework. Parts 2.1. and 2.2. give us the basis that the framework is built upon. Parts 2.4. – 2.5. cover literature that is used to develop the usability testing methodology, including the selection of the sample. Parts 2.6. and 2.7. evaluate ALM through the covered literature and suggest a framework that combines the covered literature.

2.1. Application Lifecycle Management

Application Lifecycle Management (ALM) is based on the Product Lifecycle Management (PLM). PLM is a concept related to the internal activities of a company to design and manage a product and should not be mixed with Product Lifecycle Theory, which looks at distribution of production in international trade at different levels of market saturation.

Haydaya and Marchildon (2011) tell that Product Lifecycle Management has its roots from two software innovations from the 1980s. These are the computer-aided design and the product data management system (Haydaya and Marchildon,2011). At its introduction PLM was considered through an information system approach. The "PLM system emerged to move from a set of isolated software applications toward a set of systems that interact with each other, creating a shared platform for collaboration among product stakeholders." (Haydaya and Marchildon, 2011, p. 562). The early PLM research was very business driven, and aimed at producing IS tools that businesses could use during product development to combine market research and engineering with design efforts.

From the early IS driven start, PLM has developed into a more wide concept or a mindset of looking at the wide timeframe (or the whole lifecycle) of a product already during its development. Kääriäinen (2011) explains that PLM means "the activity of managing a company's products across their lifecycles in the most effective way" (Kääriäinen, 2011, p. 3.). In the second edition of his textbook, Stark (2011) divides product lifecycle into five phases. The phases describe the evolution from an idea to the disposal of an product. These phases are 1) "When it's an idea", 2) "When it's being defined", 3) "When it's being realised", 4) "When it is in use", and 5) "When it's being disposed of" (Stark, 2011, p.2). The idea of product lifecycle management is that companies need to see product development in the framework of the whole lifespan of the product, not just the design to market phase. Based on his own research on literature, Kääriäinen states that "better lifecycle management provides benefits for companies, such as shorter time to market and improved product quality." (Kääriäinen, 2011, p. 13).

The PLM framework is the base for the Application Lifecycle Management (ALM), which takes a similar, whole lifecycle view, specific to software products. Chappell (2008) looks at ALM in from business use perspective. He defines the lifecycle of an application as the "entire time during which an organization is spending money on this asset, from the initial idea to the end of the application's life" (Chappell, 2008, p. 2). In his white paper, Chappell gives a high-level view on the management of application lifecycle. He gives three overlapping aspects to ALM. These aspects are the Governance, Development, and Operations aspect.

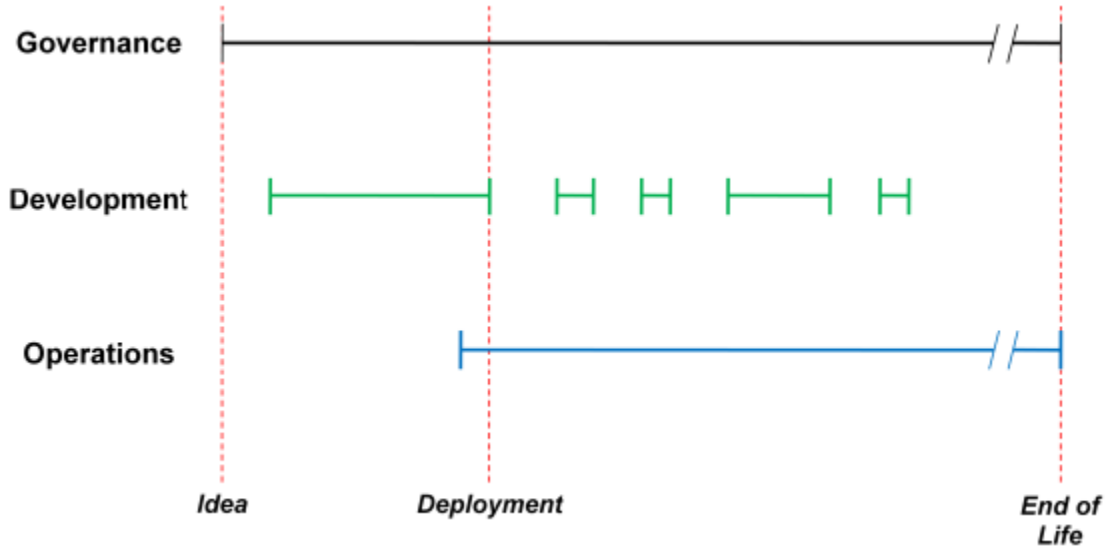


Figure 2: "ALM can be viewed as having three aspects" (Chappell, 2008, p. 2)

Governance needs to take place throughout the total lifetime of an application. It consists of project management activities, project portfolio management and application portfolio management (Chappell, 2008). Development is done in the creation phase of the application, as well as when it is updated. Development is done often in the form of projects (Chappell, 2008). Operations are the use of the application to perform its task. Operations start after the application is launched from initial development into production. (Chappell, 2008).

Understanding ALM as a mindset where software should be considered for its total existence, from initial idea to being obsolete, makes sense. This is because replacing and installing new applications in a wide scale is a not quick or cheap. It also follows intuition, that the lifecycle of

software includes the same five elements than the traditional products (ideation, defining the idea as a product, creating the product, using the product, and disposing of the product). With software the disposing of the product just doesn't need to take into account physical recycling, but rather needs for a replacing the system. However, software can be updated while in production, which adds a new layer and new level of complexity into ALM compared to the traditional PLM.

Kääriäinen (2011) takes an extensive look at ALM through five different articles and five different business cases. His aim is to build a framework for ALM. Kääriäinen ends up explaining ALM through the creation, management and interaction of artefacts that are created in different phases of the application lifecycle. He introduces six principle elements of ALM. These are:

1. Creation and management of lifecycle artefacts
2. Traceability of lifecycle artefacts
3. Reporting of lifecycle artefacts
4. Communication
5. Process Support
6. Tool Integration

(Kääriäinen, 2011, pp. 51 - 61)

The artefacts that are created during the application lifecycle can be tools that will help to create the actual end product (such as list of requirements or test cases) or actual pieces of the final product (such as source code). Kääriäinen(2011) puts emphasis on the initial design phase of the application. His research is concentrated on the development aspect, to the coordination between different artefacts (or creators of artefacts, such as developers or project managers) of ALM. Lots of the coordination can be facilitated with specific ICT products, such as testing or project management software. However, Kääriäinen (2011) does not put emphasis on evaluating the goodness of the actual artefacts.

ALM can be considered as an activity that coordinates the different aspects (governance, development and operations) (Chappell, 2008) or artefacts (products of different processes during the application lifecycle) (Kääriäinen, 2011) of the application lifecycle. Focus of the current research is on the early stages of the application lifecycle, at the initial development state. It seems that application lifecycle is sometimes used synonymously with the development lifecycle. Reason for the weighted focus on the beginning of the lifecycle might be because of PLM's heritage. When producing physical products, the initial product development often is the only product development for that product. Physical goods are rarely updated (except for some modular designs, such as desktop computers). This means that in PLM it is justified to only consider product development as a start of the lifecycle operation. However, software can be updated many times before becoming obsolete. It is important that product development aspect of application lifecycle management is considered also at the later phases of the application lifecycle.

Other current weakness of the ALM is that as a management tool its sole function seems to be facilitating communication and information sharing. This is also likely to stem from ALM's heritage of PLM. PLM was created to connect isolated systems into an interacting network of systems (Haydaya and Marchildon, 2011). Maybe a more correct terms would be Application Lifecycle Coordination. Coordination is very important and coordination is a part of good management. However, there is a need to add measuring aspects to ALM, to observer how the application is performing during its lifecycle. In order for us to truly be able to manage our applications in the long term, we need to do more than just coordinate work of different contributors. We need to be able to know how our application is doing in different phases of the lifecycle, both in the development and operational phases.

I will use ALM from a management, rather than development perspective. This is why Chappell's (2008) division to Governance, Development, and Operation is a suitable level of detail for my thesis. In order to be able to measure how an application is doing in different phases of the application lifecycle, we must next find out what should measure in an application, to evaluate its goodness.

2.2. Information System Success Model

Evaluating information systems is difficult. A fundamental challenge is choosing the right things to measure when trying to evaluate information systems. DeLone and McLean (2003) explain that there are many variables that can be seen connected to IS success from top management commitment to frequency of use of a certain system, but it has been challenging to see which variables are independent (causing IS success) and which are dependent (actual measurements of IS success). Independent factors should be improved and could be measured to predict IS success, but only measuring the dependent factors will confirm IS success.

In their article "Information System Success: Quest for dependent variable" (1992) DeLone and McLean aim to reduce the number of IS success indicators, to make further research and comparison of different researches into IS more efficient. In their seminal paper on understanding IS success DeLone and McLean covered empirical studies published between 1981 to 1987 and aimed to identify a dependent variable that would measure information system success. DeLone and McLean started by defining taxonomy of already existing success measures. DeLone and McLean identified six interdependent categories of IS success from prior research. These were the system quality, information quality, use, user satisfaction, individual impact, and organizational impact.

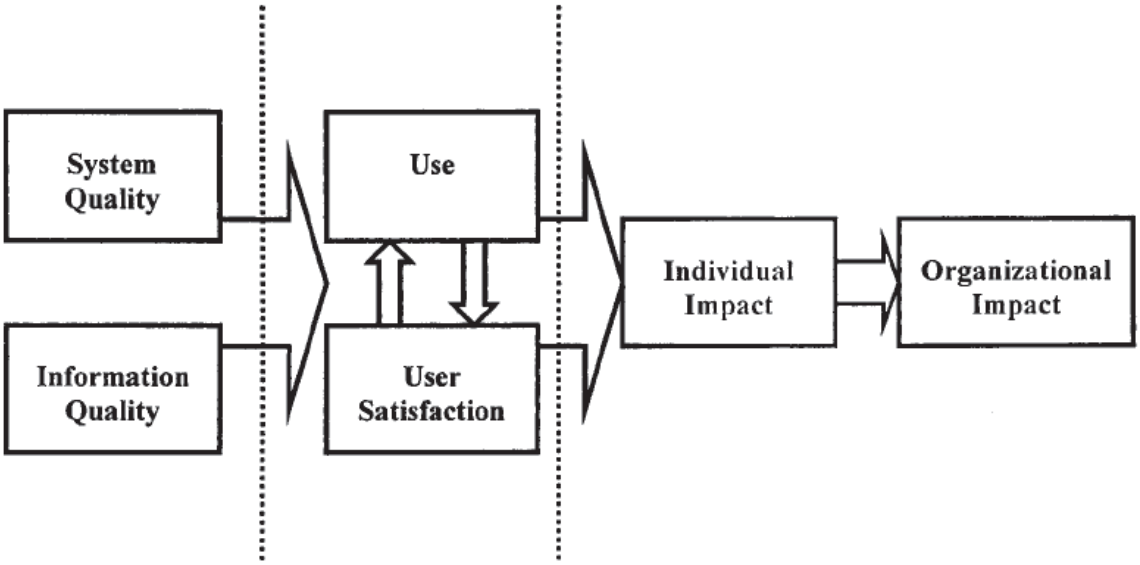


Figure: 3 "I/S Success Model" (DeLone and McLean, 1992, p. 87)

DeLone and McLean applied their taxonomy by categorizing prior research from 1981 to 1987 and conclude that "there is not one success measure but many" (DeLone and McLean, 1992, p. 88.). However, their research suggests that these success measures are interlinked and form a hierarchy. The developed model is based on the flow from data (the article uses the term "information") to organizational impact. The authors explain that system quality and information quality affect both individually and together the amount of use and user satisfaction. DeLone and McLean (1992) continue to explain how user experience (amount of use) and user satisfaction are interlinked and can have a positive or negative effect on each other. Use and user satisfaction then have an individual impact, which will have an organizational impact (DeLone and McLean, 1992, pp. 83,87.).

After its publication the DeLone and McLean IS Success Model (D&M IS SM) has been widely applied and researched. In 2003, the model had been referred to in "nearly 300 articles" (DeLone and Mclean, 2003, p. 10). The overall reception of the model has been positive. Rai et al. (2002), conducted a survey evaluating an application using the different aspects of the D&M IS SM and calculated fit statistics and estimated path coefficients for the aggregate answers of the survey. The results showed "mixed signals concerning the goodness of fit of the DeLone and McLean

structural model" (Rai et al., 2002 p.61). The fit was good for some of the paths between success elements, but not all. However, Rai et al. (2002) conclude that the D&M IS SM has explanatory power, and that an "integrated, multiconstruct dependent measure of IS success that considers beliefs, attitudes, and behaviors" seems like an appropriate way to approach evaluating IS success (Rai et al., 2002, pp. 65 - 66).

Main criticism towards the D&M IS SM comes from Seddon (1997). He claims that "the problem is that D&M attempted to combine both process and causal explanations of IS success in their model" (Seddon, 1997, p. 240). Seddon claims, that multiple meanings of the the D&M model cause "slippage from one meaning for a box or arrow to another"(Seddon, 1997, p. 242). Especially problematic is the "use" –success measure in the D&M IS SM (illustrated in Figure 3 as the "Use"-box). Seddon (1997) claims that "use" has three different meanings. These are 1) "Use" as a proxy value for the benefits of use. 2) "Use" as dependent variable of system- and information quality, an indicator of IS success. 3) "Use" as a process step in a linear user experience process flow from system- and information quality → use and user satisfaction → individual impact → organizational impact (Seddon, 1997, pp. 242 -243). Seddon explains that the problem with D&M IS SM is that it is, in his opinion, three separate models, and that presenting them as one can confuse the thought process of the user of the model, resulting into false argumentation. This is a fair and useful insight into the model. In their paper from 1992 DeLone and McLean told that they have only taken the first step in starting to narrow down the measurable IS success variables for the benefit of the whole field of study, and invite further development of their model. Seddon (1997) starts with the problem of triple meaning for "Use" and continues by suggesting a "Respecified Model of IS Success (and Use)", that he titles "Partial behavioral model of IS Use" (Seddon, 1997, pp.244 - 245).

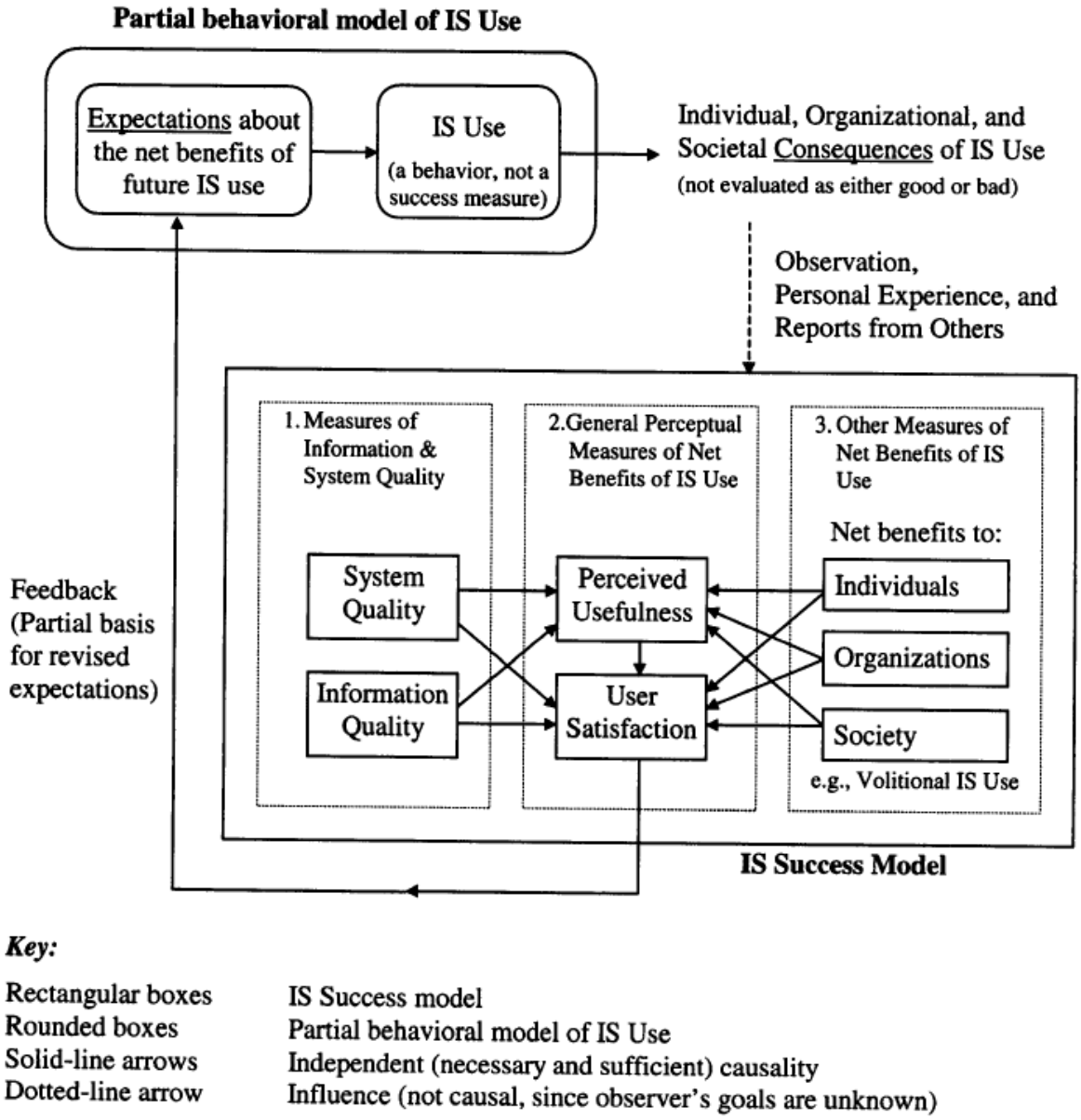


Figure 4: Seddon's Partial behavioral model of IS use (Seddon, 1997, p. 245)

Seddon's key contribution is to take away the process chart aspect of the D&M IS SM and suggest a model that describes variance relationships between different factors of IS. Seddon's model is represented in the Figure 4 above. It suggests that Perceived Usefulness and User Satisfaction are dependent variables of both Information and System Quality and Net Benefits of

IS use. Seddon also makes an important distinction between the behavior of using a system and the benefits that are caused by the use of a system. In his conclusion, Seddon (1997) highlights an important aspect, that was stressed also by DeLone and McLean (1992): there is not a single variable that can be used to measure IS success. IS success is a somewhat subjective concept that is dependent on who's opinion is being asked. Seddon (1997) explains how senior management will tend to a evaluate IS through the IS's contribution to profitability and efficiency of the organization, where the actual end users of a system could consider IS success through ease of use of the system (Seddon, 1997). Thus when evaluating IS systems, the right success measures should be chosen based on the research subjects.

In their paper that aims to update the original model, DeLone and McLean note, that not all researchers have used the model as it was intended. They tell that a lot of emphasis has been put on only some of the success factor. DeLone and McLean (2002) explain how amount of use has emerged as dominant dependent factor, even though it was intended to be only a part of the model. Lassila and Brancheau (1997) show that the amount of use is greatly dependent also on the training and support users get when introduced with a new system. In other words, use of a (especially new) system is not only dependent on the system goodness, but also on the organizational context: training and motivation for adapting the new system. As stated by DeLone and Mclean: "Simply measuring the amount of time a system is used does not properly capture the relationship between usage and the realization of expected results"(DeLone and Mclean, 2002, p. 16). DeLone and McLean (2002) also address the development suggestions by Seddon (1997). They embrace the idea of net benefits, rather than organizational impact, as organizational impact can be both negative and positive (DeLone and McLean, 2002). They also add Service Quality as one of the independent factors for IS success. DeLone and McLean (2002) discuss service quality in terms of e-commerce applications, but it fits well also for internal services (training and support), that were discussed by Lassila and Brancheau (1997). DeLone and McLean (2002) also specify the concept of use, dividing it into the "intention to use" and the actual behavior of "use".

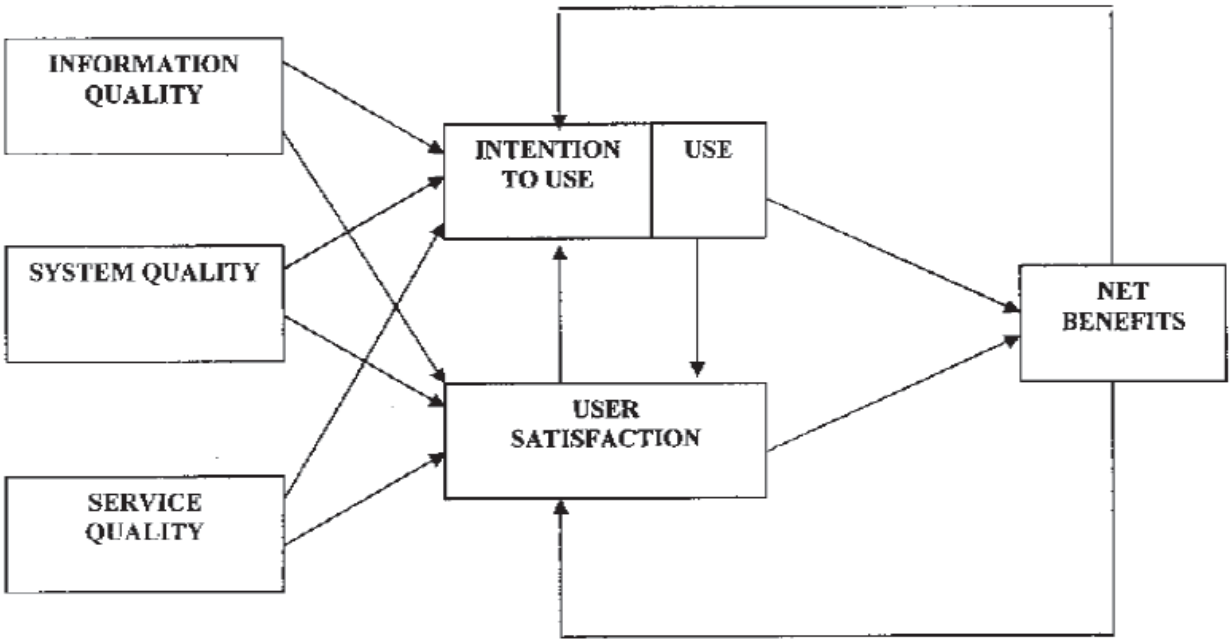


Figure 5: "Updated D&M IS Success Model" (DeLone and McLean, 2002, p. 24).

The figure above shows the Updated D&M IS Success Model. The arrows in the model "demonstrate proposed associations among success dimensions in a process sense, but does not show positive or negative signs for those associations in a casual sense" (DeLone and McLean, 2002, p. 23). DeLone and McLean (2002) advice each researcher to hypothesize the casual relationships specific to a particular study. In other words, net benefits could be seen as a dependent success factor resulting from use and user satisfaction, but also as an independent factor creating user satisfaction and increasing intention to use. Us researchers must know our users and the context to know what we are actually measuring. It is likely that for users with high commitment to an organization, as well as visibility to possible net benefits, the net benefits will have a stronger influence on future use and user satisfaction. On a final warning, DeLone and McLean reference Yuthas and Young (1998), who state that "examining satisfaction and usage measures is not an acceptable alternative to measuring performance directly"(Yuthas and Young, 1998, p. 121).

Yuthas and Young (1998) did a controlled experiment with students as proxy users to test how well usage and user satisfaction predicts actual increase in performance of completing a task with IS. They point out via their own literature review that for "usage to be a valid indicator of the system's perceived usefulness, usage must be voluntary" (Yuthas and Young, 1998, p. 119). They then create a controlled setting, where the use of IS to support decision-making is voluntary. Yuthas and Young evaluated IS success via task performance. In their specific setting the measure was "the subject's ability to minimize the combined product, holding, and stockout costs" (Yuthas and Young, 1998, pp. 118-119) for the scenario they created for the test. Use was measured "with the number of minutes a subject spent accessing computerized reports (Time) and the total number of reports accessed" (Yuthas and Young, 1998, p. 119). User satisfaction was measured with a 25 item Likert scale questionnaire. Yuthas and Young's analysis shows that the relationship between usage satisfaction and to user performance is not strong enough to "warrant their use as substitutes for one another" (Yuthas and Young, 1998, p. 121).

Based on the work of DeLone and McLean (1992, 2002), and by the extension of Seddon (1997) user satisfaction seems to be a strong dependent variable for IS success. However, Yuthas and Young show that subjective satisfaction experienced by the user is not enough. Lone and McLean (1992, 2002) and Seddon (1997) also warn against thinking that IS success is something that can be described through only one variable. It is important to take a more holistic approach, than just the subjective user satisfaction. IS success should also be evaluated through user performance, that leads to positive net benefits for the whole organization. I believe that the field of usability testing provides a set of methodologies, that take into account not only user satisfaction, but also user performance and information quality of an IS. I thus feel, that usability is an excellent measure of IS success, that could be applied in evaluating IS throughout its lifecycle. For the rest of this paper, when referring to the D&M IS SM, I am referring to the updated model from 2002.

2.3. Usability as a measure

Usability testing is not a new idea and it has been used to evaluate tools for over 60 years (Johnson et al., 2007). Kortum and Bangor make a good argument for the need of measuring usability:

> "The measurement of usability has high utility because it quantifies how well users can interact with a given product or service. Even if a product performs its primary technical function flawlessly, if a user cannot get the product to work, then that product has failed".

Kortum and Bangor, 2013, p. 68.

Usability testing is most commonly used as a design tool in software development. Usability testing can also be used as a supporting tool for decision making when deciding whether to update a system or what to update. As Johnson et al. puts it: "usability is often seen as an end-of-the-production-cycle affair" (Johnson et al., 2007, p. 320.). The experts and scholars on usability are aware that mid-development (in projects) and mid-lifecycle (in production) usability testing can have a strong positive effect on the finished product, but that usability testing in practice is still quite static or end-of-the-production-cycle (Johnson et al., 2007).

Before moving towards usability testing, we must first define what usability is. Usability is a concept that can be applied to any tool that we use. Generally, things that are easy and pleasant to use are more usable than things that are complicated and tiresome to use. To get a better definition we can look to standards. For usability, the standard is ISO 9241-11, the "Ergonomics of Human System Interaction" (International Organization for Standardization, 2013). Green and Pearson (2006) define usability, as written in the ISO 9241-11, as follows: ""The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use"" (Green and Pearson, 2006, p. 67.).

Usability is an extremely wide and somewhat subjective concept. If we wish to use usability as a dependent variable for IS success in different phases of the application lifecycle, we must keep to

a single and clearly stated definition. My research will rely on the ISO 9241-11 definition. In other words, I will treat usability as a measure that is defined by effectiveness, efficiency and satisfaction.

John Brooke explains the above-mentioned three factors of usability as follows: **Effectiveness** is "the ability of users to complete tasks using the system, and the quality of the output of those tasks" (Brooke, 1996, p. 2.). This means, that effectiveness evaluates a tool's usability by the quality of the output produced with the tool. For example, if my aim is to write a text document (a master's thesis for example), Microsoft Word is quite usable in terms of effectiveness, as indeed, a text document is being produced. **Efficiency** is "the level of resource consumed in performing tasks" (Brooke, 1996, p. 2.). Microsoft Word is more usable in terms of efficiency than an old-fashioned typing machine. The resource that is used less in this example is time (and ink and paper). **Satisfaction** is the "users' subjective reactions to using the system"(Brooke, 1996, p. 2.). All measures of usability are dependent on the user's skills and attitudes. User satisfaction is the most subjective dimension of the ISO 9241-11 definition. A user might, for example, get more satisfaction from writing a letter with pen rather than Microsoft Word. For that user, the satisfaction dimension of usability is greater for the pen than it is for the word processing program. Overall usability of a tool is combination of these three. A product might excel in one dimension and still have poor usability if it executes poorly on the rest of the dimensions.

In this master's thesis I applied usability testing to evaluating applications, but usability testing can be used to measure a variety of things. More or less anything that we *use* can be evaluated through usability. Because information technology often increases the complexity of the user interfaces, paying attention to good usability can greatly enhance an ICT-based tool's value.

Because usability is a subjective measure, it is important to use a planned methodology when measuring it. By using a controlled methodology and the same test users (or a similar group of test users), we can evaluate applications throughout the lifecycle and compare the state of the application in one phase of the lifecycle to a state in a different phase of its lifecycle.

2.4. Usability testing

There are several different tools and methods for measuring usability. Different techniques can be roughly divided into two groups: **empirical user testing** and **usability inspections** (Riihiaho, 2000, p.1). Empirical user testing is all the methods that involve test user involvement. Usability inspections can be done without actual test users of the system by usability experts (Riihiaho, 2000). No matter what technique is used, a usability tester needs to keep in mind the questions of **validity**. When conducting empirical user testing, one must also take into account matter of **reliability**.

Nielsen (1993) explains the issues of reliability and validity as follows: "Reliability is the question of whether one would get the same result if the test was repeated, and validity is the question of whether the results actually reflects the usability issues one wants to test" (Nielsen, 1993, p. 165.). There are several reasons that might decrease reliability and validity in usability testing.

Nielsen states that the main reason for reliability errors roots from the fact that users have "huge individual differences" (Nielsen, 1993, p. 166). Some people might just be more natural with information technology or have more experience with a certain system. For example testing a program on a PC-computer with test users who only have experience with Apple computers will give poor results because of weak reliability (assuming that the typical intended user for the tested system would be a PC user).

There is a way to address individual differences of test users. The problem can be reduced, if same individuals are used as test users to complete multiple usability tests. This so called "Within-Subjects Testing" (Nielsen, 1993, pp.178 – 179) addresses the errors resulting from difference in user skill levels.

"Validity is a question of whether the usability test in fact measures something of relevance to usability of real products in real use outside the laboratory" (Nielsen, 1993, p.169). Validity can be increased by performing the test with as close to real life circumstances as possible. Techniques to do this include conducting the usability tests at test users' own premises. Validity

of a usability test can also be increased by using the system's true users, compared to, for example, college students recruited to only use the program for testing. (Nielsen, 1993).

When conduction empirical user testing, it is important to take into account ethical factors. It can be quite stressful to be the observed test user. Nielsen gives practical advice on the ethical aspects on testing with human subjects. Following procedures and guidelines by Nielsen should either be articulated to the test user or otherwise taken into account when planning and conducting usability tests.

- Atmosphere of the test will be relaxed
- The test aims to evaluate the software, not the user
- First test task is designed to boost confidence
- There will be only one observer
- No information about individual test user is shared
- No information of the user's performance is shared with his / her management
- The test users are treated with respect and in a neutral manner
  (Nielsen, 1993, p. 184).

When planning my usability testing methodology to be used as part of ALM, I will aim to minimize the risk of poor validity and reliability, as well as take into account the ethical aspects of testing a system with real people. It is also important to not rely only on the subjective user satisfaction, but to measure performance as well, as articulated by Yuthas and Young (1998).

Next I will present some specific usability testing methods from prior research.

2.4.1. Visual Walkthrough

Visual walkthrough is a method that can be used to get a first impression of a system. Riihiaho (2000), in her Licensiate's Thesis "Experiences with usability evaluation methods", describes visual walkthrough as follows:

> "The participant is asked to tell what controls and groups of controls he sees in the user interface and what he thinks their functionality is. The aim of the visual

> walkthrough is to gather information of what the users notice in the interface, in what order they notice the components and in what sort of groups, how they understand the visual messages, and what terms they use as they describe the system and its functionality."

(Riihiaho, 2000 p.83).

Visual walkthrough can be done with a real system, or with a paper prototype. With a paper prototype the users can also mark things they notice on the paper. The results that can be gained from visual walkthrough depends a lot on the test user. Some people are more comfortable in saying aloud what they think, while others might not be at ease in describing what pops into their mind when just looking at a user interface. During the visual walkthrough (as with the whole test session) the examiner should stay passive and not direct the attention of test user. Visual walkthrough is thus the user "walking through" the main pages of a site or an application, rather than the conductor of the test explaining the user interface to the test user.

2.4.2. System Usability Scale

The System Usability Scale (SUS) was developed by John Brooke and was first published in 1996 (Bangor et al. 2008, p. 574). SUS is a tool that can be used to collect data from empirical user testing. Brooke himself describes that **"**The System Usability Scale (SUS) is a simple, ten-item scale giving a global view of subjective assessments of usability" (Brooke, 1996). SUS is a ten question Likert scale questionnaire. Likert scale means that the answers are given on a pre-given scale, in this case a five point scale ranging from strongly disagree to strongly agree. In his 1996 paper that introduces the now widely used SUS-scale, Brooke starts by defining usability by the ISO 9241-11 standard (usability as a combination of effectiveness, efficiency and satisfaction). He then states that "precise measures to be used within each of these classes of metric can vary widely" (Brooke, 1996). System Usability Scale does not aim to evaluate systems on individual dimensions of effectiveness, efficiency and satisfaction. Rather it scores the overall subjective experience of the user (the combination of the three). It should be noted, that even though SUS produces numeric, quantitative data, it is still subjective and dependent on

the respondents attitude towards the Likert scale. Some people might, for example, be extremities averse and avoid the 1 and 5 point answers.

Brooke advices that “scores for individual items are not meaningful on their own” (Brooke, 1996). This means that the SUS-scale produces one overall usability score of a system. The evaluator should only look at this final score, not scores about individual questions. On the next page is the actual SUS questionnaire.

System Usability Scale

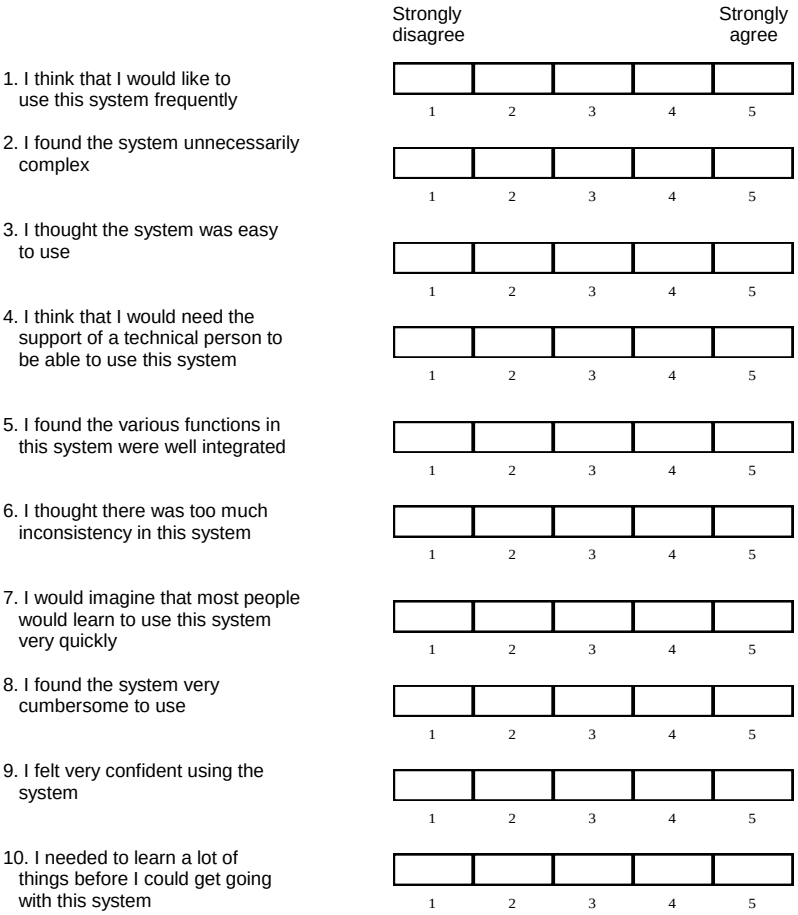


Figure 6: The system Usability Scale questionnaire (Brooke, 1996, no page)

Brooke instructs the use of questionnaire as follows:

> "To calculate the SUS score, first sum the score contributions from each item. Each item's score contribution will range from 0 to 4. For items 1,3,5,7,and 9 the score contribution is the scale position minus 1. For items 2,4,6,8 and 10, the contribution is 5 minus the scale position. Multiply the sum of the scores by 2.5 to obtain the overall value of SU".

(Brooke, 1996, no page)

Because I am developing a methodology, which will be applied to Finnish companies, the SUS questionnaire must be translated. English language questionnaire could cause error to the results (due to misunderstanding). In Appendix 1 a Finnish translation by Työterveyslaitos (no year) is presented. In calculating the SUS scores, I used an excel spreadsheet that I built to automatically calculate the SUS scores based on the answers. I highly recommend this rather than doing the calculations manually.

System Usability Scale has been widely used since its introduction. It has several benefits despite of being only a ten question questionnaire. The SUS is easy to administer to both the test user and the test conductor, it is available free of charge and has good measures of both reliability and validity (Kortum and Bangor, 2013).

Lewis and Sauro (2009) quote different research (for example Bangor et al., 2008) about the reliability of SUS. They state that "Research conducted on the SUS has shown that although it is fairly quick, it is probably not all that dirty" (Levis and Sauro, 2009, p. 2.). Levis and Sauro also confirm what the prior research on SUS had suggested by conducting factor analysis on 324 SUS questionnaires, that the reliability of SUS in measuring usability is high (Levis and Sauro, 2009, pp. 5, 7.).

SUS is also a useful tool, because it has been expanded and developed since its introduction.

2.4.3. Extensions of System Usability Scale

As mentioned above, Brooke (1996) stated that the SUS score for a system should be looked as a total. This means that the SUS score is measure of one variable, the overall usability of a system. However, Lewis and Sauro (2009) argue, through their own factor analysis, that SUS score can be decomposed into *Usability* and *Learnability* components. The questions 1,2,3,5,6,7,8 and 9 would score usability and questions 4 and 10 learnability (Lewis and Sauro, 2009, p. 5.). This additional insight might help when launching an application from development into operations. For example, if it seems that main issue with a system is the learnability rather than overall usability, this could be addressed with emphasis on training.

Bangor et al. (2008) did an extensive research on prior SUS questionnaires. They combined "nearly a decade's worth of SUS data" with "more than 2,300 individual surveys" and "200 studies" (Bangor et al., 2008, p. 575.). As a result of their extensive research, Bangor et al. produced the following figure based on the SUS scores' statistical distribution:

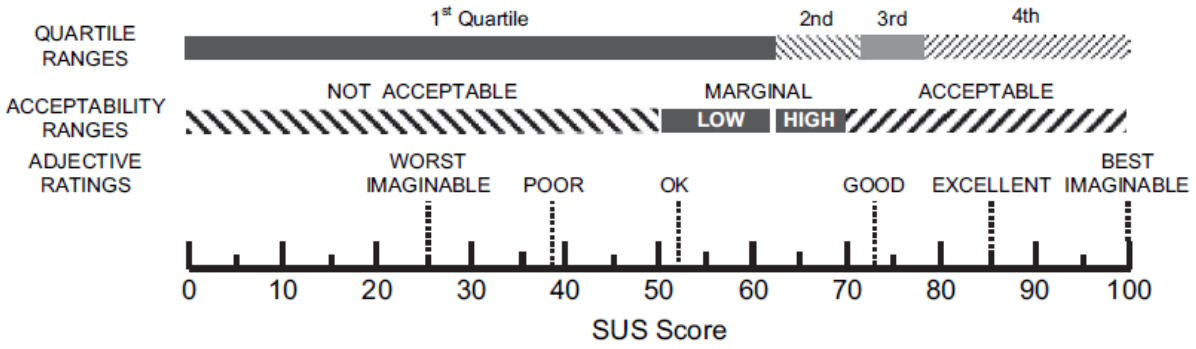


Figure 7: "A comparison of mean System Usability Scale (SUS) scores by quartile, adjective ratings, and the acceptability of the overall SUS score" (Bangor et al. 2008 p. 592)

Bangor et al. explained that products with a score of above 70 can be considered passable. Good products score on SUS on the range from high 70s to high 80s. (Bangor et al. 2008, p. 592.). The figure above shows how the best 25% of products get a SUS score higher than 80, but the "Truly superior products score better than 90" (Bangor et al. 2008, p. 592.). On the other side of the spectrum, the figure shows how SUS scores lower than 70 are on the bottom half of usability. These products with usability scores less than 70 "should be considered candidates for increased scrutiny and continued improvement" and "products with scores less than 50 should be cause for significant concern and are judged unacceptable " (Bangor et al. 2008, p. 592.).

Brooke (1996) suggested that individual SUS scores should not be compared to other people's scores, but an average SUS score of a system could be benchmarked to Bangor et al.'s framework to get a ballpark idea of the system's comparative usability.

Kortum and Bangor (2013) improved the framework by combining it with benchmarking products. The result is illustrated bellow:

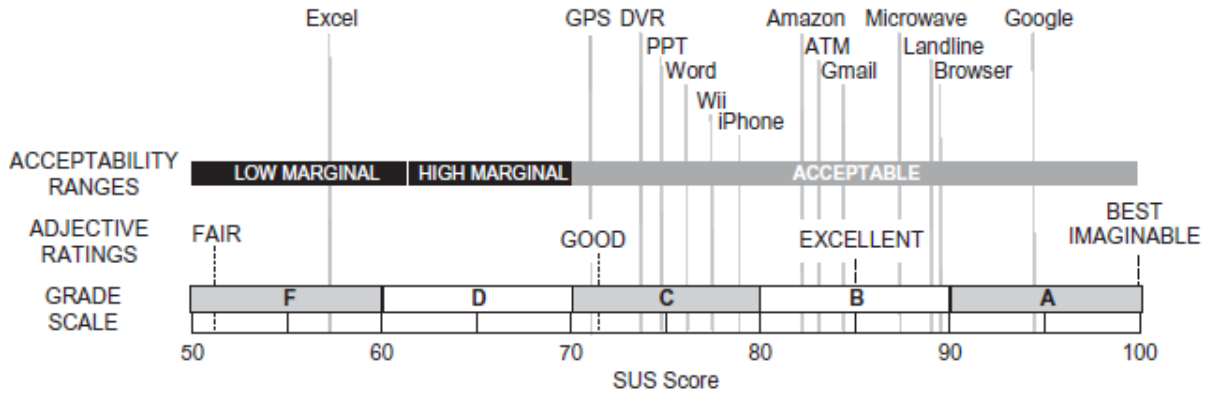


Figure 8: "Usability scores for the 14 products mapped onto the acceptability ranges proposed by Bangor, Kortum, and Miller (2009)" (Kortum and Bangor, 2013, pp. 75.)

These frameworks of Figure 7 and Figure 8 can be used when communicating the SUS scores of a usability test to managers at the governance level of ALM. This will give the SUS scores meaning for people not familiar with usability testing.

2.4.4. Expert evaluation

Riihiaho describes heuristic evaluation by Nielsen and Molich (1990) as an "informal usability evaluation method in which several evaluators comment an interface design by usability principles called heuristics" (Riihiaho, 2000, p. 30). The heuristics were defined by Nielsen and Molich as follows:

1.	Simple and natural dialogue
2.	Speak the users' language
3.	Minimise user memory load
4.	Consistency
5.	Feedback
6.	Clearly marked exits
7.	Shortcuts
8.	Good error messages
9.	Prevent errors
10.	Help and documentation

Figure 9: "Ten usability principles, *i.e.*, the heuristics (Molich & Nielsen 1990)" (Riihiaho, 2000)

Molich's and Nielsen's list of heuristics is a very good tool for usability evaluation. However, an expert user of a system, or an expert in usability testing will be able to identify at least the major usability problems without a checklist of heuristics.

Riihiaho (2000) explains expert evaluation as an inspection of a user interface that is done based on the user's expertise rather than the pre-defined heuristics. In ALM context the expert evaluation is a good tool, as it just means getting feedback on usability aspects from the expert users of the system. These people are already likely to be involved in the development aspect of ALM as at least testers, so their involvement in usability evaluation will probably be quite easy and will not generate any additional costs.

2.5. Technology Acceptance Model

In order to create useful usability testing methodology for ALM, we need to address the limitations of usability testing. The major difficulty in measuring usability is that the perceived ease of use is a subjective matter. Users are people and each person has different attitude and skillset when it comes to using technology. In order to be able to conduct usability testing, we should understand a little more about the sources of that subjectivity. In the context of IS usability, we can look at different users' different attitudes towards technology. D&M IS SM suggests variables that indicate IS success. Technology Acceptance Model and its extensions aim

to explain how to achieve the behavior of use, which is a crucial process step towards organizational net benefits in D&M IS SM.

One of the most prominent theories addressing differences between users is the Technology Acceptance Model (TAM) by Fred Davis, first introduced in 1989 (Venkatesh and Davis, 2000). Davis (1989) starts his seminal paper from the problem that even though IT could enhance performance of white collar workers, the worker's unwillingness to adapt new systems can nullify possible performance gains (Davis, 1989). Davis moves on to explain, based on prior research, that perceived usefulness and perceive ease of use seem to be two major determinants of IT acceptance (Davis, 1989). Main contribution of paper by Davis (1989) was to suggest two measurement scales (in form of questionnaires) to measure that perceived usefulness and perceive ease of use of IS.

From the identification of perceived usefulness and perceive ease of use as major factor affecting technology acceptance, the model has been extended on several occasions to include more factors that affect technology acceptance. Venkatesh and Davis (2000) updated TAM to include social influence processes (such as voluntariness of use) and cognitive instrumental processes (such as job relevance and output quality) and formed TAM2.

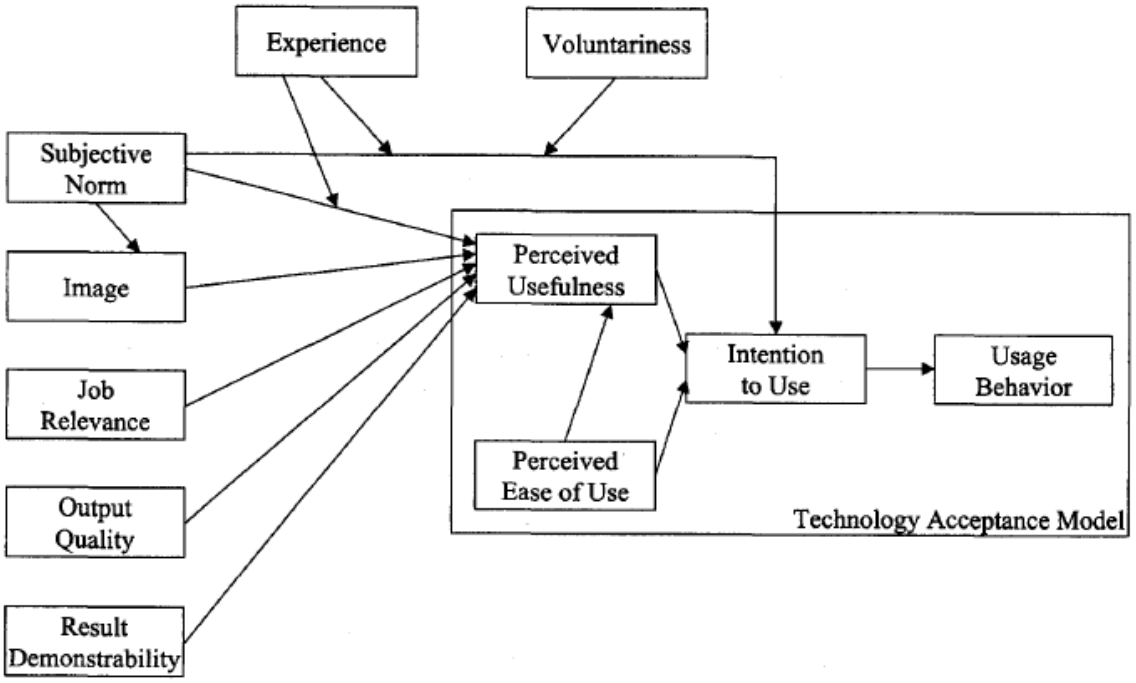


Figure 10: "Proposed TAM2 – Extension on the Technology Acceptance Model" (Venkatesh and Davis, 2000, p. 188.).

TAM2 identifies factors that affect the perceived usefulness of a system. Important for work systems is the subjective norm, the outside pressure or orders to use a system (Venkatesh and Davis, 2000). Subjective norm is "strong prior to implementation and during early usage, but weaken over time as increasing direct experience with a system provides a growing basis for intentions toward ongoing use" (Venkatesh and Davis, 2000, p.190). TAM2 shows how even with subjective norm (mandatory use of a program), users tend to make own choices about technology use based on their own experience. Mandatory use does not thus neglect the need to achieve IS success.

TAM2 model illustrates that intention to use a system is a combination of perceived usefulness and perceived ease of use. Perceived usefulness can be connected to the effectiveness and efficiency aspects of ISO 9241-11 definition of usability. Perceived ease of use can be connected to the satisfaction aspect of ISO definition of usability. In this way we could argue that the same

factors that affect the intention to use would also affect the usability test scores, for example SUS scores.

Bagozzi (2007) criticizes TAM, especially the assumption that intention to use actually leads to use of a system. He also criticizes how TAM is being updated in a "patchwork"-manner complicating it more and more, even though there is a problem in the core assumption of intention equals use, that the whole model is based on (Bagozzi, 2007). However, even if TAM as such could not predict actual use, extensions of TAM do provide important insights into the demographic factors that could affect attitudes towards technology. Being aware of these factors is important when selecting a sample for usability testing.

Venkatesh and Davis (2000) identify that experience with a system has an effect on the intention to use. In their 2003 paper Venkatesh, Morris and Davis aim towards a Unified Theory of Acceptance and Use of Technology and state that in addition to experience, also voluntariness, gender and age are significant moderating factors for intention to use a system (Venkatesh et al. 2003). Usability testing is subjective, but TAM models suggests underlying factors that statistically are likely to affect attitudes towards technology. If usability testing is conducted in long term as part of ALM, the sample of test users should be heterogeneous in terms of experience, gender and age. Similar selection of sample in terms of experience, gender and age should be used throughout the lifecycle of an application, even if the exact same users cannot be used each time. This is not a substitute for within-subjects testing, but allows for a bit more control over results of usability tests over lifecycle of an application.

So far we have established that user satisfaction is an important independent success measure of IS, but is not sufficient alone (DeLone and McLean, 2002). Use of net benefits is problematic, as depending who we are using as research subjects, it might act as a dependent or independent factor. Yuthas and Young (1998) approach the matter differently, prompting to measure user performance. Usability testing does not directly measure net benefits to organization, but it can measure effectiveness and efficiency of use of system. Usable applications are not only nice to use, but will also make work more efficient, which is likely to lead to net benefits to organizations in terms of increased productivity.

SUS by Brooke (1996) measures users' perception of a systems' usability, which includes satisfaction, efficiency and effectiveness. Depth can be added to the general SUS score by conducting visual walkthroughs and expert evaluations, which aim at finding user attitudes and factors that can hinder efficiency and effectiveness of use.

There is a problem with reliability in usability testing, stemming from individual differences in user skills and attitudes (Nielsen, 1993). Reliability can be controlled by having a heterogeneous sample in experience and demographic factors identified by Venkatesh et al. (2003).

In previous chapters we have identified what and how data should be collected from the operations and the development aspects of the ALM. Next I will consider research that will help us to determine how the data should be used for decision making on the governance aspect of ALM.

2.6. Governance aspect of ALM

Chappell (2008) defines the governance aspect of ALM as business case development, project portfolio management and application portfolio management. Governance also includes application management, "deciding when updates and larger revisions make business sense" (Chappell, 2008, p. 3). Kääriäinen (2011), mentioned process support as a key element of ALM. Kääriäinen (2011) suggested a framework to evaluate ALM solutions, or technical tools for ALM, but he raised an important point, that support for the actual development process, or development project, is an important aspect of ALM (Kääriäinen, 2011). Chappell (2008), does state that development cycles are an important part of ALM, but does not mention project management as such as part of ALM. To be useful base for framework for usability testing throughout the application lifecycle, Chappell's model needs to be modified to take into account the different layers of managers: the strategic managers and the project managers. I thus suggest splitting the governance level into strategic- and project management levels. The strategic level, involves the project portfolio management (including the choice and evaluation of possible outside vendors), and application portfolio management, as well as the evaluation of project and IS success (including the decisions to update or abandon applications). Project management level

of the governance aspect, in my extended model based on Chappell (2008), is concerned with business case development and defining and managing the scope of development projects, as well as other project management tasks.

2.6.1. Strategic management level of the governance aspect of ALM

The strategic level of governance aspect consists of managing several applications, or several application lifecycles simultaneously. Strategic management of several applications is called application portfolio management (APM). Simon et al. (2010) work towards a framework of APM and based on their literature review, combine prior research to form their own definition:

> "Application Portfolio Management is the ongoing application of systematic and structured decision-making processes to evaluate an organization's applications along various dimensions (from a business and a technical viewpoint), weigh various actions for the purpose of optimization, and implement appropriate actions to resolve identified issues and meet key enterprise objectives. The promise of Application Portfolio Management lies primarily in reducing the complexity of the application landscape, which is approached from a holistic viewpoint."

Simon et al., 2010, p. 38

The definition is somewhat lengthy, and Simon et al. (2010) explain it in pieces. APM is an ongoing activity, that should be systematic. The end result of APM should be the reduction of complexity of the application landscape of an organization (Simon et al., 2010). Chappell (2008), in his model of ALM, only superficially considers APM, but does portray the same idea of need for simplicity. Chappell takes a cost driven view to APM, explaining that the goal should be to "avoid duplicating functions across different applications."(Chappell, 2008, p. 3). Simon et al. (2011) develop a model, where APM consists of four stages. These are: 1) Data Collection, 2) Analysis, 3) Decision Making and, 4) Optimization. (Simon et al. 2010). Simon et al. (2010) state that APM addresses the "general lack of a permanent approach to the application landscape that draws on structured and rational methods for making decisions about application investments on a portfolio-wide basis." (Simon et al., 2010, p.39). I embrace this notion and

extend it to be true also for ALM. Decision makers need a "permanent approach" to measure IS in order to be able to compare long term performance and evaluate different applications over time. Usability testing could be part of the systematic approach in the "Data Collection" phase of Simon et al. (2010) model. Simon et al. (2010) looked APM from cost benefit approach. Usability testing does not fit directly to their model, as it does not take into account actual development costs, and can also predict benefits indirectly through user satisfaction and user performance. Usability testing will thus not alone be enough to produce all the data needed for APM, but it could be a part of the approach, as usability testing could predict net benefits (or lack of them) via measuring user performance. APM consists of more than just internal analysis in an organization. APM "may also facilitate the selection of applications to be outsourced as well as the request for proposals"(Simon et al., 2010, p. 39). This leads us to the second important function of the strategic management level of the governance aspect of ALM.

Part of the "Optimization" -stage of APM, after "Data Collection", "Analysis" and "Decision Making", is the possible vendor selection (Simon et al., 2010). If development of an application is outsourced, companies need to engage in vendor management. This starts with selecting the vendor. After vendor selection a contract needs to be written. Li and Yetton (2009) summarize prior research by stating that "outsourced IT projects are governed by both formal outsourcing contracts and post-contractual relationships or psychological contracts" (Li and Yetton, 2009, p. 47). However, as any project, outsourced IT projects carry a risk of failure. The formal outsourcing contract is not always enough to produce a successful application. As Ilan Oshri stated in an interview by Flinders: "failure happens in major IT outsourcing simply because both client and vendor are keen to get down to business instead of doing their homework properly" (Flinders, 2011, p. 11.). Businesses need tools to better evaluate their vendors, as well as tools to help the vendors better understand their customers. The problem with IT projects is, that it is very difficult to define what actually is needed. The problem of setting good project scopes is discussed in the next part 2.6.2. The implications to both project management and vendor contracts is discussed in the end of part 2.6.2. as the problem of setting good scopes, and the tendency of scopes to change during projects is crucial to both the strategic and project management levels of the governance of ALM.

2.6.2. Project management level of the governance aspect of ALM

Software development is commonly done through projects. In the ALM the first development phase (creation of the application) is the most critical one. Later changes will only present incremental changes to the initial application. The development and testing of the updates is also commonly carried out in project format. Kääriäinen and Välimäki wrote about ALM for distributed software development and state that "The purpose of ALM is to provide integrated tools and practices that support project cooperation and communication through a project's lifecycle."(Kääriäinen and Välimäki, 2011, p. 182).

Achieving project success in ICT development projects is difficult. The traditional approach is to consider a project a success if it meets its predetermined goals. The project goal is to achieve "a desired future state" (Artto et al., 2011, p. 22), while keeping within the project restraints. The project restraints are traditionally described as the triple constraint of **Time**, **Scope**, and **Cost** (Artto et al., 2011, p. 23.). Time and cost are relatively objective figures. They can be measured in quantitative terms, even though there are challenges in getting accurate measures of spent time and identifying hidden costs. Time and cost are goals that are related to the project budget. Lee et al. (2012) investigated the effect of cost and schedule goals to project escalation and state that "the perceived success of a software project depends to a great extent on whether or not it is delivered within budget and on schedule" (Lee et al. 2012, p. 57). Schedule and cost goals indicate how company resources are used in reference to the project plan, but give no reference to the quality of the outcome of the project. They do not describe well the *desirability* of the achieved *future state* after the project outcome is launched into production.

Artto et al. (2011, p. 17.) mention that a fourth requirement, **quality,** does exist. However, often the only measure that we can use to evaluate the quality of the project outcome is to evaluate if the pre-defined triple constraint was met. Cost and staying on schedule describe how the actual development process went compared to the initial plan. Only scope goal measures the final product, the characteristics of the application. The new version of an application can have many benefits compared to the old one. It can be developed within budget, on time, and as specified in

the beginning of the project. However, if the users do not accept the product, a technically successful project might still be considered a failure.

Artto et al. explain the scope as the "technical, functional, and qualitative features of the product resulting from the project" (Artto et al., 2011, p. 23.). Defining these technical, functional and qualitative features can be challenging. Lewis, in an interview by Finders, tells that "even big companies, lack internal resources to specify technical and operational requirements fully or at all" (Finders 2011, p. 12). The same idea is presented by Vasiljeva (2012), based on her years of project experience, as follows:

> "The art to identify, clarify and conclude business requirements, but, the most important, to elicit and extract required functionality, restrictions and constraints, capture, communicate and specify them in written way sometimes seems practically unachievable goal."

Vasiljeva, 2012, p. 43.

Defining the scope can be seen as an art (as Vasiljeva, 2012, puts it) and it is a challenge in our increasingly project driven economy.

In addition to being difficult to set, scopes often change during projects. Sometimes this is a positive thing. For example if the change in the scope is a response to a change in the environment. Sometimes the changes in the scope can be a result of failure in project management or scope creep. Scope creep refers to an unwanted phenomenon where "scope of services expands beyond items covered in the contract document" (Bellenger, 2003, p. 58.).

Companies are not blind to project success or failure when they see it. Even if, in the planning phase, project goal is formally defined via time, cost and scope, customer satisfaction (be it internal or external) can be used in evaluating project success (Artto et al., 2011). However, if you are in the customer side of the project (as companies in ICT development projects often are), the vendor can reject your objections about a non-satisfactory end product, as long as the predefined scope is met. Because setting scopes for projects is so challenging, maybe additional

contractual goals are needed for vendor management. This would encourage the suppliers (or vendors) of ICT development projects to use their expertise also in setting the scope.

The vendors should have the expertise to be able to help in setting the scope, but as long as they are allowed to be evaluated only based on scope given by the customer, there is little incentive to do so.

In terms of ALM strategic managers can use the classical project management goal of meeting the triple constraint as an initial measure of project success. In vendor management, usability metrics (such as a minimum acceptable SUS score) could be articulated in the formal contract to encourage vendors to be more pro-active in helping customers to set very good scopes.

At project management level, there is need for tools for project managers to help with setting of the scope of the project. This means deciding what the program should do and how it should interact with the user. In internal development projects, usability testing can be used to find out desirable functions for the new application to be articulated in the project scope. During large projects, usability testing can be used in mid-development to confirm that the project is on the right track, and that the developed functionalities are actually going to improve the satisfaction and performance of the users.

2.7. Conceptual framework

Before trying to evaluate if usability testing could be a useful tool for ALM, I present a framework for usability testing throughout application lifecycle based on prior research.

From the prior research I learned that ALM is a complicated task. There are several actors within and outside of the organization that need to be taken into account during the application lifecycle. Currently the ALM research and product development of ALM software tools concentrates on coordinating the actions of different actors (such as the framework by Kääriäinen, 2011). These actors can be, for example, product managers, project managers, developers, testers, line work managers, and the end users of a system. The different actors operate with different aspects of the ALM. We can call these the governance, development and production aspect (Chappell,

2008). However, I feel that the governance level has two levels: the strategic management level and project management level. The justification for the two levels is that project managers need only concern with the execution of a project, where strategic managers need to evaluate several projects, manage a portfolio of applications, and engage in vendor management. In other words, under the strategic level of the governance aspect, there is simultaneously several applications with their unique lifecycles. The lower levels (Project Manangement, Development, Production) only need to consider one application at a time. A single person in an organization might play several roles in the model. A person might be a project mangaer, who still has strategic responsibilities, such as vendor management. For the sake of the model, these functions are still separated even if carried out by the same person.

ALM already has tools to connect and coordinate for example software development and project management (Kääriäinen and Välimäki 2011). However, there is not a clear common measure of goodness of an application that would guide all the actors involved in the ALM, from the strategic governance level, to the operational line work level.

Different factors are important to measure on different aspects of ALM. DeLone and McLean (1997, 2002) and Seddon (1997) have worked towards dependent variables for IS success. However, the factors that they identified are still quite wide concepts that guide researches in choosing the actual concrete measurement points. I suggest usability testing outputs to be fitting measurement points for some of the major success variables in the D&M IS SM. Usability testing could also be used to improve some of the independent success factors via better design.

In Table 1 bellow, the first column combines different aspects of ALM with real business challenges or goals related to that aspect. Second column shows an IS success variable that could be used to measure or explain success on that aspect of ALM. In the third column is a suggested output of usability testing that could be used to measure or improve each success factor.

Table 1: How usability can measure IS success throughout the application lifecycle		
Aspect of ALM (Adapted from Chappell, 2008). With a desired effect of usability testing to a real business challenge.	**IS success variable (from DeLone and McLean, 2002)**	**Aspect of usability measure corresponding IS success variable**
Governance: Strategic management Decision making would be improved with data from usability testing. Usability testing would answer to the need for "permanent approach to the application landscape" from Simon et al. (2010).	Net Benefits to organization interpreted as a dependent variable, which measures project-, application-, or vendor success.	Aggregate usability evaluations of applications collected over time. Data used to compare applications and vendors of applications.
Governance: Project management More likely project success, when problems of identifying good project scopes (Vasiljeva, 2012 & Finders, 2011) mitigated with usability testing as a part of requirements identification.	Net Benefits to organization interpreted as an independent variable, where successful projects are likely to create more use and more net benefits to the organization.	Objective measures of usability evaluations (task performance, work efficiency)
Development Higher system quality with input collected during iterative software development. Usability evaluations can be used to prioritize backlog and smaller updates post-development.	Improved Information and System quality (independent variable)	Usability evaluations used to evaluate prototypes mid-development and guide future development.
Operations Usability focus during development will increase user satisfaction, effectiveness and efficiency of the system users.	User Satisfaction (dependent variable)	Subjective measures of usability evaluation: SUS score.

The desired results listed in the first column will not automatically be realized just from conducting usability testing. Rather, the benefits will stem from improved decision making with data from usability testing. The benefits can only be realized, if the available data leads to better actions. Uniform and controlled usability testing methodology allows for use of aggregate usability data for strategic decision making, such as project-, application, and vendor portfolio management.

It should be noted that the division on the table is a little bit repetitive. The development and operations rows are actually included already in the strategic- and project management rows. improved information quality will cause user satisfaction and net benefits to organization. User satisfaction data from operations will be used for the decision making on the governance level. T

To make this less confusing, and to illustrate the connection of different aspects of ALM, I will present the same ideas in terms of the timespan of ALM. The framework is based on the simple, but nicely descriptive model by Chappell (2008).

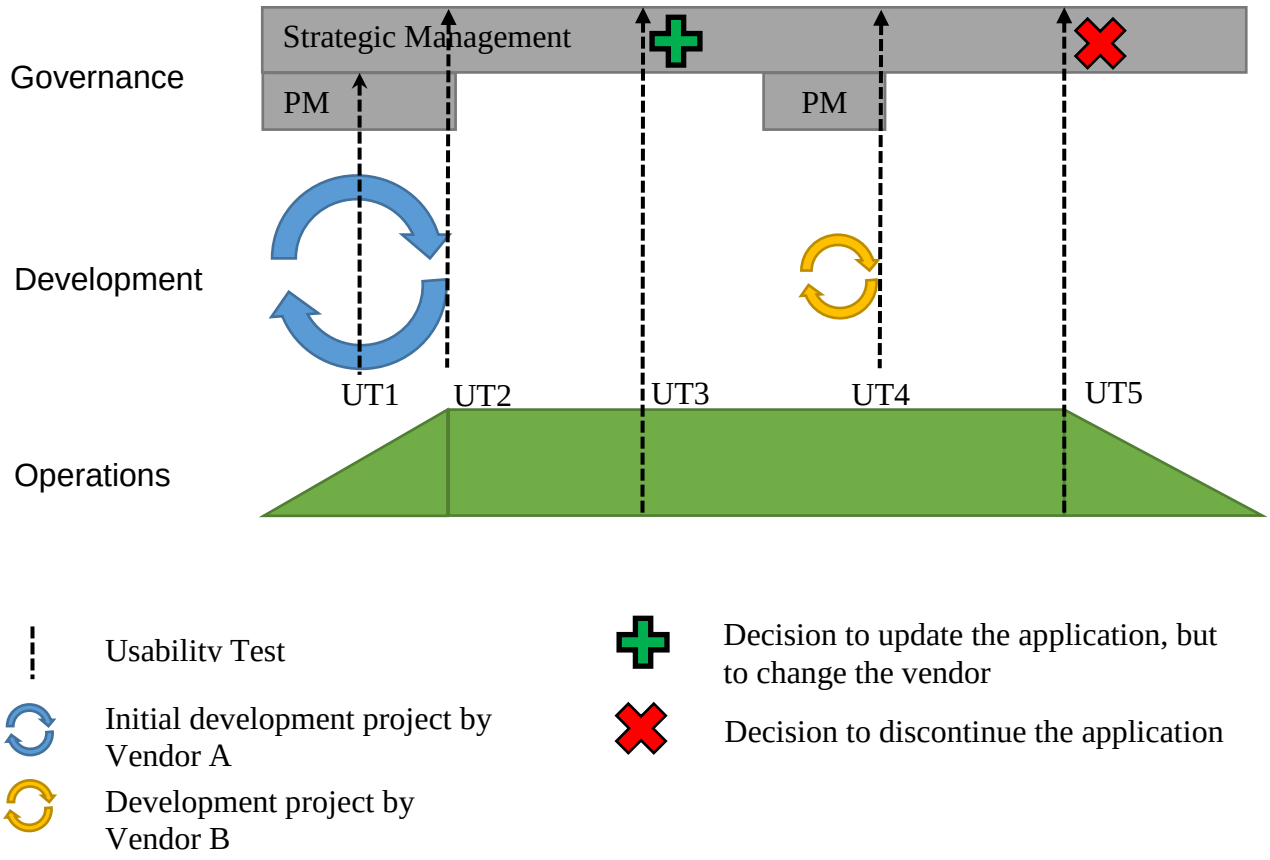


Figure 11: Conceptual framework

The figure above shows the conceptual framework that is built around Chappell's (2008) model. It shows two dimensions. The vertical axis shows the different aspects of ALM. The aspects are also often managed or carried out with different departments of an organization. In many cases, the project aspect is outsourced to an outside vendor. The horizontal axis shows time, or the total application lifecycle from the initial development to termination.

The conceptual framework illustrates occasions on which usability testing would be beneficial for ALM. The arrows UT1 – UT5 illustrate usability test occasions. The arrows show where data is collected from and what level of governance should use the data. This illustrates better than

the table 1, that usability testing as a part of ALM is a process of collecting data from operations and development to improve decision making on the governance level.

In the framework, projects are assumed to be conducted in an iterative manner. Size of the circle represents the size of the project. Project should start with a usability test to help define the scope of the project (project management row of Table 1). Let's assume that this application will replace an outdated one, and that the final usability test of the legacy system was base for the scope definition of this application. With large projects, it is advisable to conduct usability testing with prototypes throughout the development cycle. This is illustrated with UT1. Mid-development feedback will allow for changes even during the projects (development row in Table 1). The results from mid-development should not be used at the governance level. After the project, when the product has been launched from development to operations, usability testing should be conducted to evaluate project success by comparing the usability of the new and the old system (strategic management row in Table 1). These test occasions are illustrated with UT2 and UT4. End-of-project tests can be used to evaluate vendors, if development was outsourced, or to evaluate project teams, if development was in-house (also strategic management row in Table 1). The data from end-of-project tests can support decisions whether or not to launch the developed application from project organization to operations. The test users should be borrowed to the project organization from operations. If the project organization has real users, for example user acceptance testers, these would be suitable people to act as test users for UT1, UT2 and UT4.

The managed application should also be tested on regular intervals in operations (operations row in Table 1). This is illustrated with UT3 and UT5. Usability testing in operations produces aggregate data that allows strategic decision makers to compare applications and vendors of applications. The green cross illustrates a decision, where usability tests (complimented with other metrics, such as realized project cots) prompt for updating of the application, but with a different vendor from the initial development project. Usability of the application in operations might be so poor, that it needs updating. The organization could decide that the updating should be done with other vendor, than the one who supplied the sub-optimal product in the first place.

There can be limitations in the different forms of vendor lock-in, but it is assumed in this example, that switching is possible.

The data from continuous usability testing can be used to make decision to update, abandon, or change the applications. If application's usability significantly weakens, even when changes to the application are not made, it might indicate that the application is no longer suitable for the changed environment. This might cause a decision to discontinue the application as illustrated with the red cross in Figure 6.

Through this framework, I suggest that usability testing could be used as a unified measure to enhance the management of different aspects of ALM by different decision makers. Common focus on usability would give a common goal to the separate actors involved with ALM. It is important to stress, that I don't suggest that usability testing could equal ALM. Decision makers still need to take into account cost of development and other requirements than just usability. Such requirements might be, for example, security issues. Even with added dimension of measurement, the communication between different application stakeholders remains as an important function of ALM. Major focus of ALM should still go to facilitating communication and managing organizational change to accept and train the use of new applications.

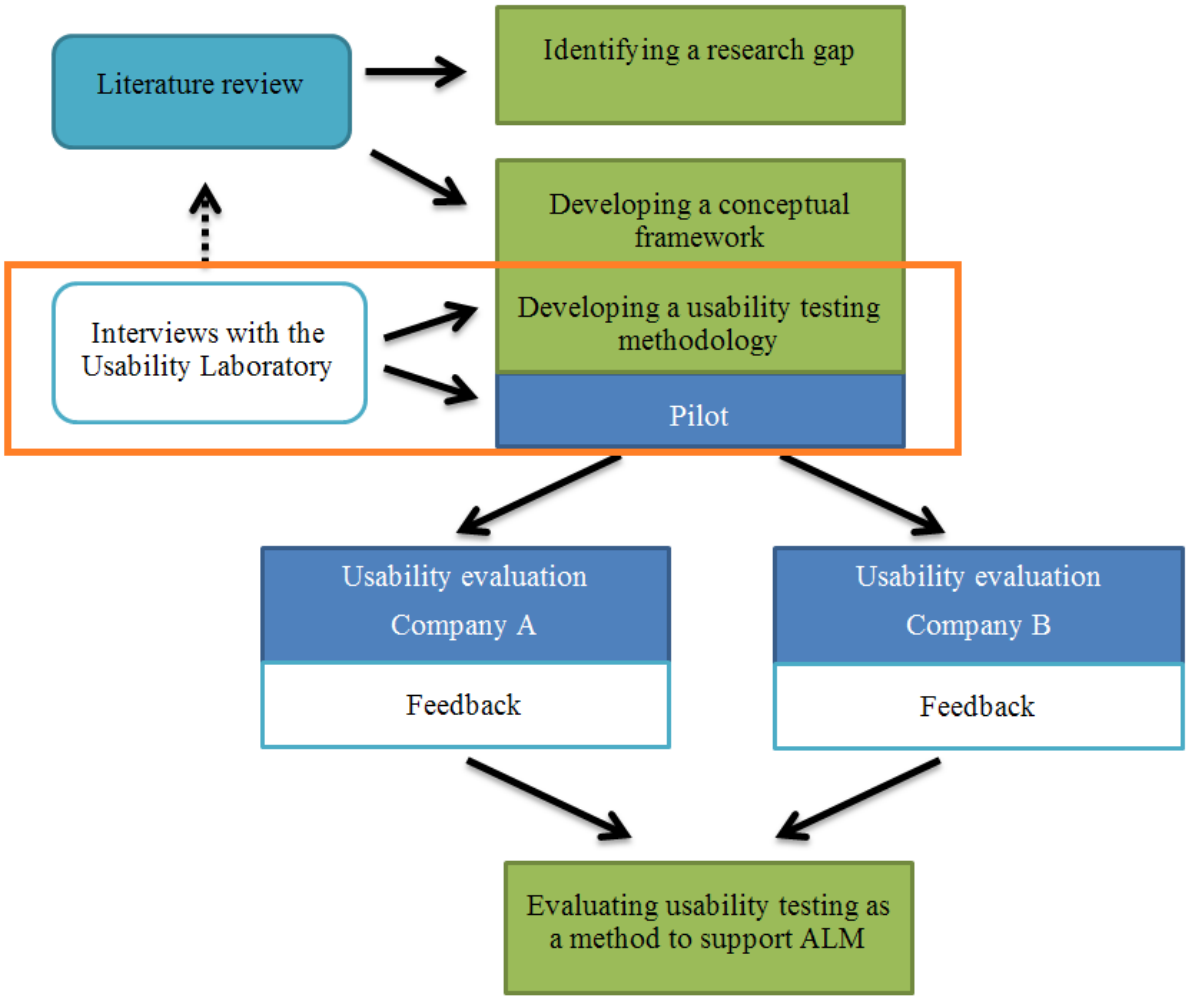
However, the framework above illustrates, that conducting regular usability testing will not only benefit the developers of applications, but also enhance decision making on the governance aspect of ALM. One uniform usability testing methodology, and a few experts of usability testers in an organization, could greatly improve ALM. Same usability testers should be involved in testing of all of the organizations applications. ALM is after all not the management of one application (which is illustrated above), but the management or governance of several applications.

Based on the literature review, I feel that there is promise for usability testing as a part of ALM, as it measures both user satisfactions, and user performance. Crucial assumption for usability testing's use as a part of ALM is that usability testing can produce aggregate data that is comparable over long term. This requires a controlled usability testing methodology. Next chapter describes the development of a possible usability testing methodology and aims to

answer the first research question: **What kind of usability testing methodology could be used to evaluate IS throughout the application lifecycle?**

After which we will move on to the empirical part of this research addressing the second research question: **Will company representatives see the developed usability testing methodology as a useful tool for ALM at different phases of the application lifecycle?**

3. METHODOLOGY



In order to test the developed framework, two levels of methodologies needed to be developed. The first phase was to develop a usability testing methodology that was applied to the company projects. This allowed for a controlled representation of usability, even when dealing with separate companies. A baseline methodology also allowed for feedback and development of the methodology after the data collection phase, hopefully resulting into something that can have real applications in future projects.

The second phase was to develop a set of questions that was used to evaluate the usability testing methodology and usability testing's suitability for ALM.

3.1. Developing the usability testing methodology

Usability is a subjective measure with a rather wide definition. In order to use usability metrics for strategic decision making in ALM, the concept of usability has to be somehow controlled. This allows for comparison of aggregate data over time. This is why a well-defined usability testing methodology, including sample selection, must be developed. Also, usability testing's applicability as a ALM tool can be evaluated with separate company projects only if the actual idea of usability is controlled in both projects. As mentioned before, in this research, the usability evaluation conducted with the developed methodology act as a proxy variable for the concept of usability testing. The testing methodology should be comprehensive enough, to allow it to represent usability testing in general. Secondary goal of this research is to suggest a testing methodology that businesses (not only academic scholars) would find useful for ALM. The methodology should thus be relatively quick and straightforward to apply, as it is likely that companies will not be willing to spend extensive resources to in-house usability testing.

The basis for the usability testing methodology was formed during meetings with the Aalto University School of Science's Usability Laboratory. In our first meeting 6.11.2012 Riihaho, Kaipio, Haanperä and Rämänen instructed me towards some relevant readings. After this I did some research of my own. The literature review of this paper summarizes the literature that was used in developing the usability testing methodology. The final form of the methodology was formed during the follow-up discussions with the Usability Laboratory 20.11.2012, 26.11.2012 and 7.1.2013. The methodology was then pilot tested in the premises of the Usability Laboratory in Otaniemi 26.11.2013 with the Development Manager of Company A.

3.1.1. Discussions with the Strategic Usability Group

I gained valuable insight into usability testing during the four discussions with the Strategic Usability Group. In addition to giving suggestions for readings, Riihaho, Kaipio, Haanperä and

Rämänen shared some of the best practices that they had found through their own research. They suggested, as a part of the usability testing, to ask users the following questions:

- What do you like about the system?
- What would you like to add to the system?
- What do you not need or would remove from the system?

I also gained input from the members of the Strategic Usability Group to finalizing the background question form (see Appendix 2). Riihaho, Kaipio, Haanperä and Rämänen suggested finding the test user's relationship towards information technology by asking about web usage on free time. The background form has dual purpose. It aims to relax the test user by asking easy questions, and collects data about age, experience with the system and relationship with information technology. These factors, as explained with Technology Acceptance Model, can have an effect on views of the new system.

From their previous experience, members of the Strategic Usability Group told that a suitable number of test users is five. This is a surprisingly small number, but they explained that it is enough to find the most pressing issues with the system. After five test users, there is a diminishing return on new insights than can be found with each participant. However, as the complexity of a evaluated system increases, it is appropriate to add more test users.

In our discussions I learned, that a common technique in usability testing is to give the user tasks and observe when the tasks are being performed. This is a good way to find problems that the user might not be consciously aware of, or cannot articulate. It also shows the efficiency of the system as a number of errors made during the test. Effectives of the system can be evaluated through whether or not the user can complete the task at all. Members of the Strategic Usability Group instructed that it is especially important not to intervene or instruct the user during the "performing tasks" -phase, so as not to guide or direct the user.

Most importantly, the members of the Strategic Usability Group stressed to take into account the ethical aspects of testing with real users. It is very important to articulate to the test user (or participant) that usability testing is about the program, not about the user. It is also important to

keep all the results from individual user anonymous and state this to the user. Usability testing should not be used to evaluate users even as part of ALM, in order to keep organizational attitudes towards usability testing open and positive.

3.1.2. Pilot test with the development manager of Company A

After developing the first version of the methodology, it was tested with the development manager of Company A. The test was conducted so that the development manager and I were in a room with a computer and tape recorder. Members of the Strategic Usability Group were behind a two-sided mirror and observed the session via the mirror and microphones in the room.

Piloting the methodology provided some valuable insights. The actual testing methods seemed to be suitable, but there was room for improvement in how to conduct the test. Most notably, prior to the test, it should be confirmed that all the equipment and programs function. During the pilot test, I had trouble with usernames and passwords, as one of the systems logged out during the usability test. I also learned, that it is a good idea to practice the test a couple of times before conducting it. It is not good if the expert is nervous when observing users. One nervous tick that I had, was to use overly formal language. It is important to make the user feel relaxed and comfortable, so natural language (or level of formality that the test user is comfortable with) should be used. In addition, the expert conducting the evaluation should give as little feedback as possible when the user is using the system. The user must be allowed to struggle with possible problems, as this can give valuable insight about the system that is being evaluated. The task list for observation was also too short in the initial test methodology, and did not allow for proper interaction with the new system prior to the SUS questionnaire. More tasks were added for the final version of the methodology. The test tasks should take at least 20min of time to perform (including comments and questions by the user).

An additional benefit of the pilot test, especially if the evaluated system is used by external stakeholders (such as customers), is to present the methodology to the company whose customers are being interviewed during the usability testing.

3.1.3. The developed usability testing methodology

The table below represents the different methods and questionnaires that form the developed usability testing methodology. It is a combination of techniques from prior research and discussions with the Strategic Usability Group. The order of the tasks is important, for not to prime the user with positive or negative aspects of a program prior to SUS questionnaires. Table 2 shows also approximation of duration of each step as well as the rationale behind each step.

Table 2: The developed usability testing methodology

Time	Task	Resource	Rationale
10 min	Introduction and background questions.	Background question form (Appendix 2).	To relax the test user. Background questions can later be used to explain possible outliers in the results.
5 min	Open the old system and ask the user to demonstrate typical task with the system.	Old system open.	To get to know the user better. To bring the old system to the user's mind before SUS questionnaire.
5 min	SUS questionnaire for the old system.	SUS questionnaire (Appendix 3)	Get a score that can be compared with the score of the new system, as well as the literature.
10 min	Ask the three questions: • What do you like about the old system? • What would you like to add to the old system? • What do you not need or would remove from the old system?		These can give the main usability successes and usability problems of the old system. Good discussion opener for the user. The questions are somewhat loaded (especially last one has a negative bias). This is why they should be asked after SUS, so that SUS score is as objective as possible.

5 min	Open the new system. Visual walkthrough of the new system (user explains what she thinks different elements on the user interface mean. Conductor of the test stays passive).	New system open	Evaluate first impression of the new system. Gives insight into the layout and navigation of the system.
20 min	Give user tasks to the user. Observe usage of the system.	Printed tasks for the new system.	To find usability problems that the user is not aware of. To give the user experience in using the system before SUS questionnaire.
5 min	SUS questionnaire for the new system.	SUS questionnaire (Appendix 4).	Get a score that can be compared with the score of the old system, as well as the literature.
5 min	Ask the three questions: • What do you like about the new system? • What would you like to add to the new system? • What do you not need or would remove from the new system?		These can give the main usability successes and usability problems of the new system. Good discussion opener for the user. The questions are somewhat loaded (especially last one has a negative bias). This is why they should be asked after SUS, so that SUS score is as objective as possible.
5 min	Ask for any final or closing comments about the new system. Ask how the user feels about switching to the new system. Thank for participating.		To leave a good feeling about the interview. Final comments can reveal the most pressing issues or findings for that user.

Times in the above table are approximations. Different users can be more vocal in different phases of the session, which will affect the times. Conducting the whole test takes about an hour

(+/- 15 min, depending on the user). It is advisable to reserve at least 1h 15min to have a little bit of slack, if something unexpected occurs.

When conducting usability testing, it is important to keep in mind the ethical aspects of usability testing. Namely keeping individual information confidential and anonyms as well as being friendly and supportive towards the test users (in limits of not guiding their usage during the test).

When planning the usage tasks, it is important to work with users or experts of the system, so that the tasks are natural for the users. Instructions should be given in writing. This minimizes the interaction with the user and the tester during the task and hopefully helps the user to concentrated on the task, rather than the test situation. The instructions should be clear. Pictures were a good addition to the instructions.

The above-described usability testing methodology is a collection of prior methodologies, but it does introduce a new approach to the SUS scale. In this methodology, I combine within-subjects testing with the SUS questionnaire. This means that same test users complete the SUS questionnaire for both the new and the old system. What is interesting in the SUS results is not the absolute SUS scores (which are subjective) but the average of differences between the SUS scores of the new and the old system. This means that individual differences in perceiving the Likert-scale of SUS can be controlled even with a small sample size (five test users per usability evaluation).

In the printed material, I color coded SUS questionnaires for the old and the new system (See Appendices 3 and 4). This was useful when conducting the testing, as they are quite similar and it is important not to mix them.

3.1.4. Format of the usability evaluation report

Results of the usability tests were presented to the companies in the format of a written document and a presentation that summarized the results.

The format of the written report and presentation was as follows:

1. Overall results and main findings of the usability evaluation
2. Introduction to the testing methodology on a general level
    - Definition of usability
    - Visual walkthrough
    - Performing tasks
    - SUS score
3. Presenting SUS scores in a table comparing the score of the new and the old system.
4. SUS score of old and new system presented with data from Bangor et al (2008).
5. Usability successes with illustrations (individual functionalities of the application).
6. Usability weaknesses with illustrations
    - Each weakness is presented with a suggestion for improvement
    - Suggestions are presented in three groups of severity: Critical, Often repeated problems, and Minor problems
7. Other comments from the users that might be useful / interesting to the company / project management
8. Summary of the main findings

With Company A, the report included also additional data based on customer feedback, that was outside the usability testing presented above.

3.1.5. Selecting the test users

The core idea of the above-described testing methodology is that it allows for comparisons on different test occasions during the application lifecycle. However, experience with a system has a major impact on attitudes towards a system. This is demonstrated, for example, with the Technology Acceptance Model. Venkatesh and Davis (2000) state, that when user experience with a system increases, so does the importance of user's own acceptance of the technology rather than the subjective norm. On the other hand, without experience it is difficult to form any strong opinion on a system (which is why subjective norm is stronger in the beginning). If a test user is seeing the system for a first time during the usability test, the SUS score and attitude towards the system might not reflect the system's real usability. We are naturally change

resistant, and if the previous version of the application has served us well, we might be skeptical towards an update. After all, changes require always some amount of learning and extra effort by the users. So even with Within-Subjects Testing (same users evaluate both new and old versions of the system) reliability of the results can be weak.

This means that if possible, test users should include as wide range of experience as possible. In other words, it is advisable to use also other's than the most eager and tech savvy users as test users. In addition, the sample should have representation from different age groups and both genders. Even if it is not possible to use the same test users throughout the applications lifecycle, a heterogeneous demographics and experience should be aimed for.

3.2. Interviews with the company representatives

To evaluate if usability testing could be used as a ALM tool, I interviewed company representatives involves with the two ICT development projects that I conducted usability evaluations for. Focus of the empirical part is on the development phase of the application lifecycle, as this was the part where the usability tests were conducted. The interviews were conducted after presenting the usability evaluations. The interviews were semi-structured. The guiding questions are presented next.

First set of questions aimed to test whether there is a real business need for better ALM tools in the development phase. Focus is on the scope goal of project management, as this is something that usability testing could help to define.

1. How did you define the scope of this project? What about previous projects?
2. Had you in this, or previous projects, had trouble with defining the scope?
3. If so, can you describe the problems you have had with the scope goal?
4. Have you had trouble with vendors when evaluating the results of projects?

The next set of questions was aimed to identify the interviewees' prior experience with usability testing.

5. Have you conducted usability testing before?

6. What is your opinion on collecting input from ordinary users versus ideating with experts?

The next question asks specifically about to the used testing methodology.

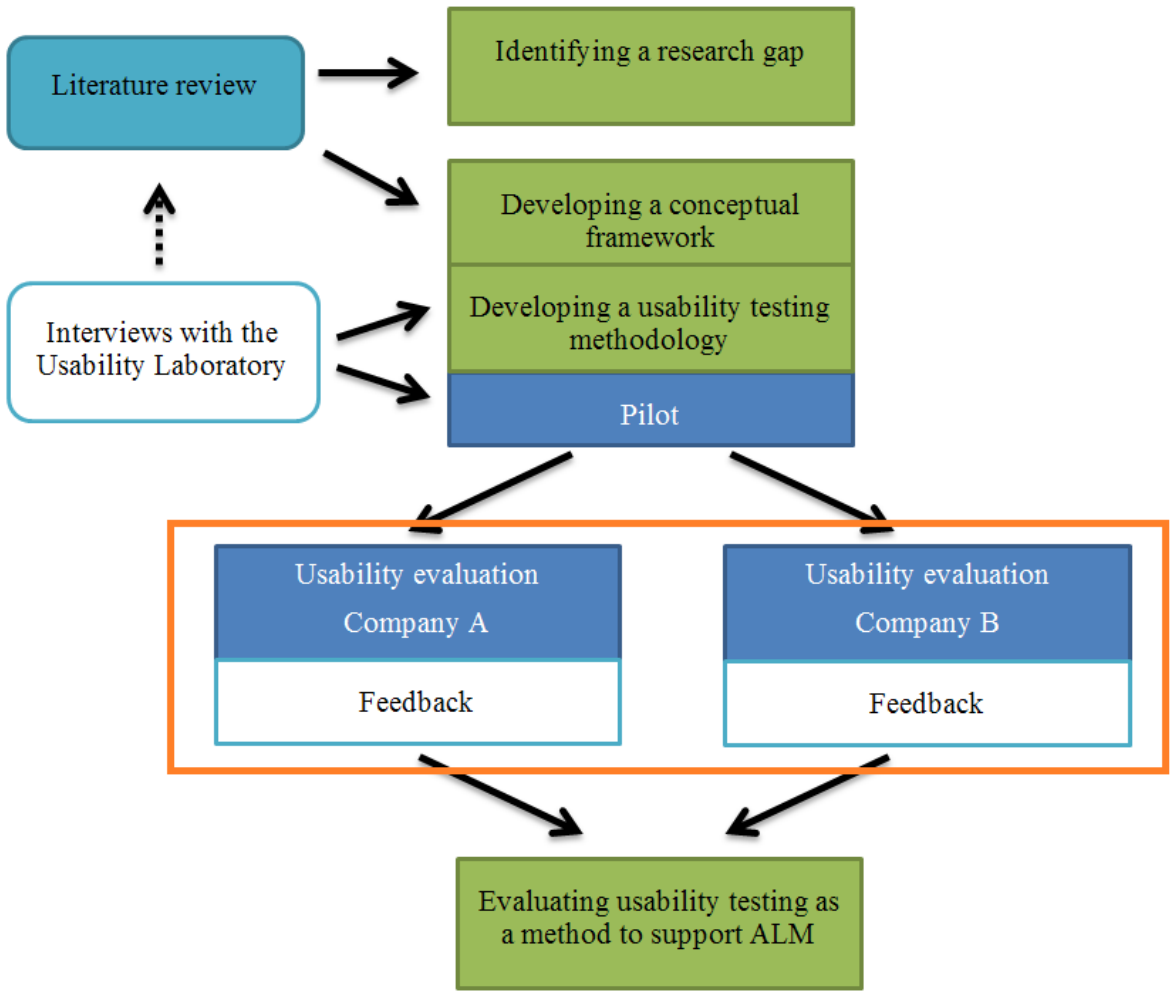
7. How did you find the end result of this usability evaluation? How did you find how the report was structured? Was the report too long? Was the information on the methodology redundant?

Final set of questions tests usability testing's suitability for different functions of ALM.

8. Could you see usability testing in the future as a tool for defining the scope?
9. Could you see results of a usability evaluations as part of contracts towards vendors as one of the formal project outcome requirements?
10. Could you see usability testing to be used continuously during the product lifecycle as a tool for continuous improvement?
11. Would you yourself have time and/or willingness to conduct the usability evaluations, if the methodology was well documented?

In addition to these generic questions, the discussion is likely to touch on project specific matters. These are confidential and outside of the scope of this research.

4. RESULTS



4.1. Conducting the usability evaluations

In December 2012 and April 2013 I applied the developed usability testing methodology to two separate company projects. This resulted in a usability evaluation report for both companies. The companies and projects were independent from each other. After presenting results of the usability evaluations to the respective companies, I interviewed company representatives (including the project managers) in order to evaluate whether or not usability testing could be extended to be used as a ALM tool. For company A, a demo of the new system was evaluated in

mid-development. For company B a finished version already in production was evaluated. For both companies the new versions were benchmarked against the previous versions of their respective systems.

4.2. Interviews with the company representatives

Below are the results of the interviews with the two companies.

4.2.1. Company A

Company A's ICT development project was to update a web based sales channel for information. The sales channel is business critical for Company A and usability of the system is important in creating a good customer experience. The sales channel is used by experts in Company A's client organizations.

I conducted the usability testing with five different test users, all from different client companies of Company A. The usability testing was conducted in test users' own premises. The tests were conducted by using a demo of the new system. The test users saw the system for the first time during the usability testing.

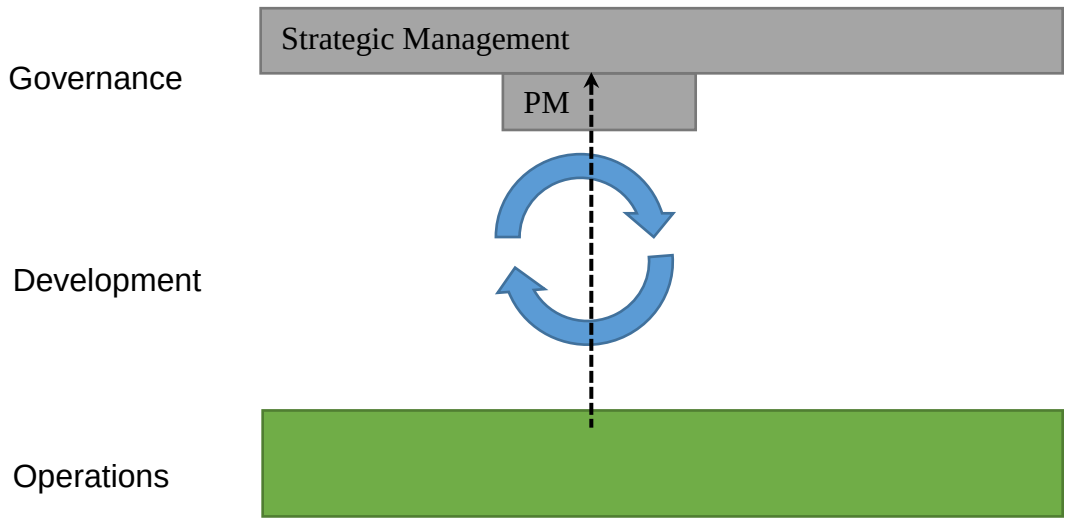


Figure 12: Usability test for Company A in the conceptual framework

Figure 12 illustrates where this particular usability test (UT:A) fitted in the developed conceptual framework. The test was done to gain insights to which direction to develop the new application, but also to evaluate the vendor of the concept. This means that the usability evaluation gave information for both project organization (PM level) but also about the vendor, which falls to vendor management of the strategic management level. The project was mixed, with the concept work outsourced, but the actual development done internally.

In spring of 2013 I interviewed Company A's development manager and project manager who were in charge of the development project. Together they had over 20 years of experience in different ICT projects. Interviews were held in Finnish and included some project specific information. Below is summarized the discussion that is relevant for my research, and without any confidential information.

1. How did you define the scope of this project? What about previous projects?

The company representatives told that for this project, defining of scope was done in a group with experts. They have used user involvement in defining the scope in some previous projects.

2. Had you in this, or previous projects had trouble with defining the scope?

The company representatives did not state any specific problems with this project. They explained that they are getting better at defining scopes as a company. "Experience helps" told the project manager of Company A.

3. If so, can you describe the problems you have had with the scope goal?

The technical manager explained that "Some of the challenges we have experienced have been in-house". There has been some challenges with internal communications. For Company A not much development work is outsourced, so they did not report problems with evaluating projects with vendors. The company representatives did mention that they have experienced some challenges with specific technical specifications.

4. Have you had trouble with vendors when evaluating the end results of projects?

As mentioned in the previous question, Company A representatives did not identify any significant problems with vendors. They told how they have long lasting and stable relationships with their vendors. The consultants even work at Company A's own premises.

5. **Have you conducted usability evaluations before?**

Company A has used usability evaluations now and then in the planning phases of projects.

6. **What is your opinion on collecting input from ordinary users versus ideating with experts?**

The company representatives told how the customer satisfaction and thus the user input is for them the most important thing. However, they explained that there are also benefits in using experts in defining the scope. Customers are not always clear on what they want or cannot articulate their needs very well. The project manager and development manager both agreed that the best thing is to use both customer and expert input.

7. **How did you find the end result of this usability evaluation? How did you find how the report was structured? Was the report too long? Was the information on the methodology redundant?**

The development manager liked how the new system was benchmarked against the old system. This was a concrete way of seeing that progress was made (the new system scored better than the old one). The company representatives also liked to see how the SUS score compared to the wider data by Bangor et al. (2008). The company representatives also explained, how using numeric figures in the results fits well with their company culture. They stated that using the SUS scores makes the results seem more quantitative. The SUS also democratizes the test users, in that each test users opinion is equally presented with the SUS score. During the interview we discussed, that a major concern about usability testing was, that the sample of test users was small and people have different personalities. This way if only open comments of users were used, some peoples' opinion might have dominant role in the results.

The project manager liked how "the report started with the usability successes". She explained how people working with, and especially in charge of, projects can feel protective over their work, and it can be unpleasant to have someone point out only the failures. Nature of usability evaluations is to focus on the areas of improvement, but it is good idea to start with usability successes, to make the audience more open. The development manager added that seeing also the successes "confirms some of the big decisions that we made about the design". In this project, there was some quite large changes to the old system, and there were some worries how customers would react to them. The company representatives were pleased to see that some of the changes they made, had been noted in the usability successes section.

Additional suggestion for the reports was to have a prioritized list of the usability issues in a condensed form in the appendix of the report.

Project and development manager told that it was ok that the report also includes information on the methodology in addition to just presenting the results.

The illustrations in the report showing the discussed features of the application was well received by the company representatives.

8. Could you see usability testing in the future as a tool for defining the scope?

The Company A representatives felt that usability testing is useful for design of a new system, but the "threshold to involve customers is high" (Development Manager, Company A, 2013). For systems that have internal users, the threshold to conduct usability testing would be lower. The project manager stated that "usability testing should probably not be used in every project". She continued to explain, that it is probably too much effort if only small incremental changes are made. However, both agreed, that for bigger changes it is a good idea to conduct usability testing in the planning phase.

9. Could you see the result of a usability evaluations as a part of contract with the vendor as one of the formal project outcome requirements?

Company A did not have problems with interpreting the scope with their vendors. They also did a lot of development in house, so a further contractual clauses was not required.

10. Could you see usability testing to be used continuously during the product lifecycle as a tool for continuous improvement?

The Company A representatives felt that this should be evaluated "one project at a time" (Project manager, Company A, 2013). If usability testing would be used throughout the product lifecycle, a good pace would be once a year. For customers the pace should be even less frequent. Customer interviews could be done every two years.

11. Would you yourself have time and/or willingness to conduct the usability evaluations, if the methodology was well documented?

Both representatives of Company A agreed that they would not want to themselves conduct the usability testing. "It sounds like a good idea, as long as it isn't me who has to do it" (Development manager, Company A). Company representatives did state that there could be need for more of overall lifecycle management over products. If there was a person who would have overall responsibility over the lifecycle management of a software, that person could be responsible for the usability testing.

If there was aggregate data of a software over its lifecycle, spanning over updates, it could be used for future definitions of scope and requirements.

Other comments

Development manager of Company A stressed that doing usability testing with customers is also communication towards the customer. He stated, how "there is a risk if we ask a lot of feedback, but don't implement customer suggestions to the system". When using customers as test users, it is important to note, that not only do we get information from the customer, but the customer also forms an opinion and expectations over the new system. If the expectations are not met, this will have a negative effect on customer satisfaction.

4.2.2. Company B

Company B's application development project was to update a web based key information system used by the organization's own experts. The updated information system is used for customer service and a wide range of back office tasks. The usability testing was conducted with five test users. Compared to Company A, Company B's system was only in internal use. This way the usability of the system does not have a direct effect on the customer satisfaction. However, a more usable system could increase employee satisfaction as well as the efficiency of customer service and thus have an indirect effect to the customers. The new system had already been implemented a month prior to the usability testing. The old system was still up and running, and all of the test users used the old system on side with the new system (the new system still lacked a few critical functionalities). However, compared to Company A, the test users did have more usage experience with the new system.

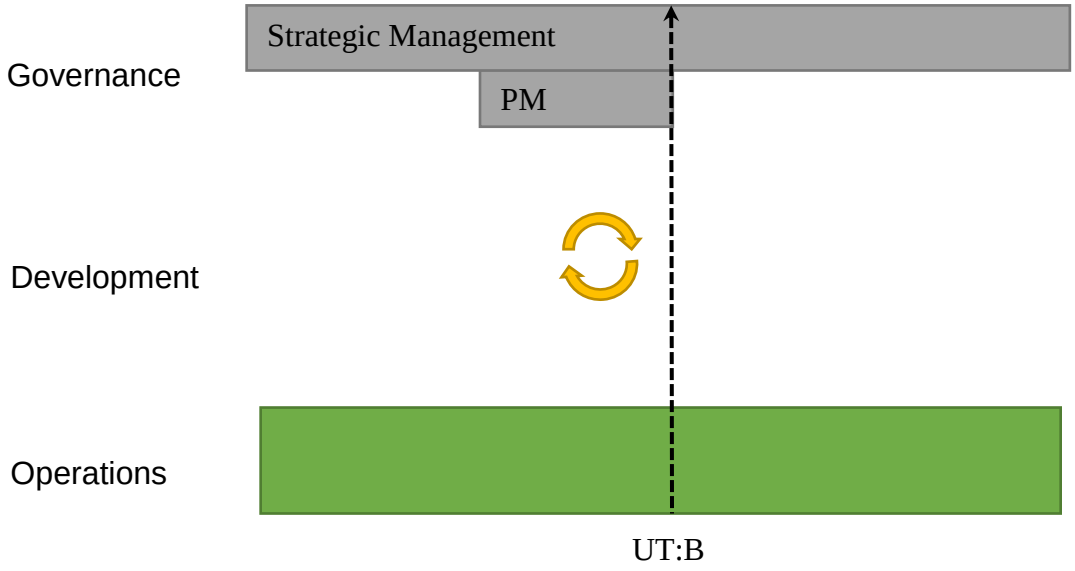


Figure 13: Usability test for Company B in the conceptual framework

Figure 13 above shows the usability test for company B (UT:B) in the conceptual framework.

For Company B, I interviewed the project manager and a system specialist for the development project. The project manager had ten years of experience in project management. The system specialist had seven years of experience in ERP-system development, coding and defining

requirements. Like with Company A, the interview was held in Finnish. Below is summarized the discussion that is relevant for my research, and without any confidential information.

1. How did you define the scope of this project? What about previous projects?

Project manager of Company B explained how for this project, the vendor offered to help to define the requirements via user inspections. One reason for choosing this specific vendor was their reputation with designs with good usability. It was also important, that the old functionalities worked in the new system. In prior projects, usability testing was not used and the scope definition was more expert driven.

2. Had you in this, or previous projects, had trouble with defining the scope?

The company representatives stated that they have experienced challenges with defining the scope.

3. If so, can you describe the problems you have had with the scope goal?

The project manager and system specialist stated that there often are conflicting goals for new systems or system updates. There are also challenges when making changes to legacy or background systems. In this particular project, there were also internal problems with communication. In addition, the project manager of Company B told how "specialists are not always very skillful in specifying requirements". Both company representatives agreed that it is best to derive the requirements straight from the users, as was done with this specific project.

4. Have you had trouble with vendors when evaluating the end results of projects?

Company B was doing this ICT development project with agile software development methods. The company representatives explained that "in agile we had not experienced much trouble with the vendor in regards of the scope". This was contrasted to the "traditional development method, where you wait 4 months from ordering to the end result". In traditional method there has been some challenges with evaluating the end results with vendors. The company representatives

explained that long relationships with vendors are better. As the project manager put it: "If they understand your business, they are better at delivering what you need".

5. **Have you conducted usability evaluations before?**

Company B had not used usability evaluations of finished products before.

6. **What is your opinion on collecting input from ordinary users versus ideating with experts?**

Project manager told that end user input is better and often faster. Especially good thing about user input is that there is no argument against user needs. This means, that if the specifications are done based on expert understanding, there can be conflicts between two or more experts who are basing their arguments on their own views. With user input, there can be data to back up arguments and resolve conflicts about different functionalities.

7. **How did you find the end result of this usability evaluation? How did you find how the report was structured? Was the report too long? Was the information on the methodology redundant?**

The company representatives thought that it was a good idea to start with the usability successes. This made it easier to be responsive for improvement suggestions. They also liked the numeric representation of SUS scores. However, they did not think that the comparison based on data from Bangor et al. (2008) was especially useful. The company representatives would have appreciated a more specific benchmarking to the usability of programs in the same industry.

The company representatives agreed that it is ok to have the usability issues prioritized on a general level into three severity categories. A further prioritization within the three levels was not required. What the company representatives did suggest was to prioritize with cost, if possible. This means both cost of the usability issues and cost estimation of fixing it. This would "Make it more understandable for management what to fix cost efficiently" (Project Manager, Company B, 2013). The company representatives continued about the cost aspect by telling how even small changes can often cost a lot to implement. They explained, that even if there are

usability issues, if there is a workaround that people can use, the business side can often live with the usability issue.

Company B representatives were not that excited about the comparing nature of the usability evaluation. They told, how the first evaluation of the old system should be done in beginning of the project. Second evaluation should be done later after implementing the project. When the project is over it is "time to move away from the old system already" (Project Manager, Company B, 2013).

8. Could you see usability testing in the future as a tool for defining the scope?

Company B representatives answered that usability testing could be used as a tool for defining scopes of future projects. They also stated, that this depends on the complexity of the system to be updated (or created). For something very simple, it is not necessary to put extensive resources in user testing. The system specialist of Company B brought up the great variance of users. Both company representatives agreed, that when conducting usability testing, differences between old and new users needs to be taken into account.

9. Could you see the result of a usability evaluations as a part of contract with the vendor as one of the formal project outcome requirements?

The project manager answered that usability testing should be used "maybe not as a tool of validation, but as a tool of development". She stated that it does not feel like a good idea to but usability metrics as requirements to a contract with vendors. Exception was, that if a totally new interface was ordered, maybe then also usability metrics could be used in the requirements.

The company representatives told that problem is that usability and functionality can be conflicting. For example, some security features might be essential for a system, but hinder usability. This makes it difficult to demand from vendors. Other problem is the subjectivity of usability testing. All the clauses in a contract should be objective.

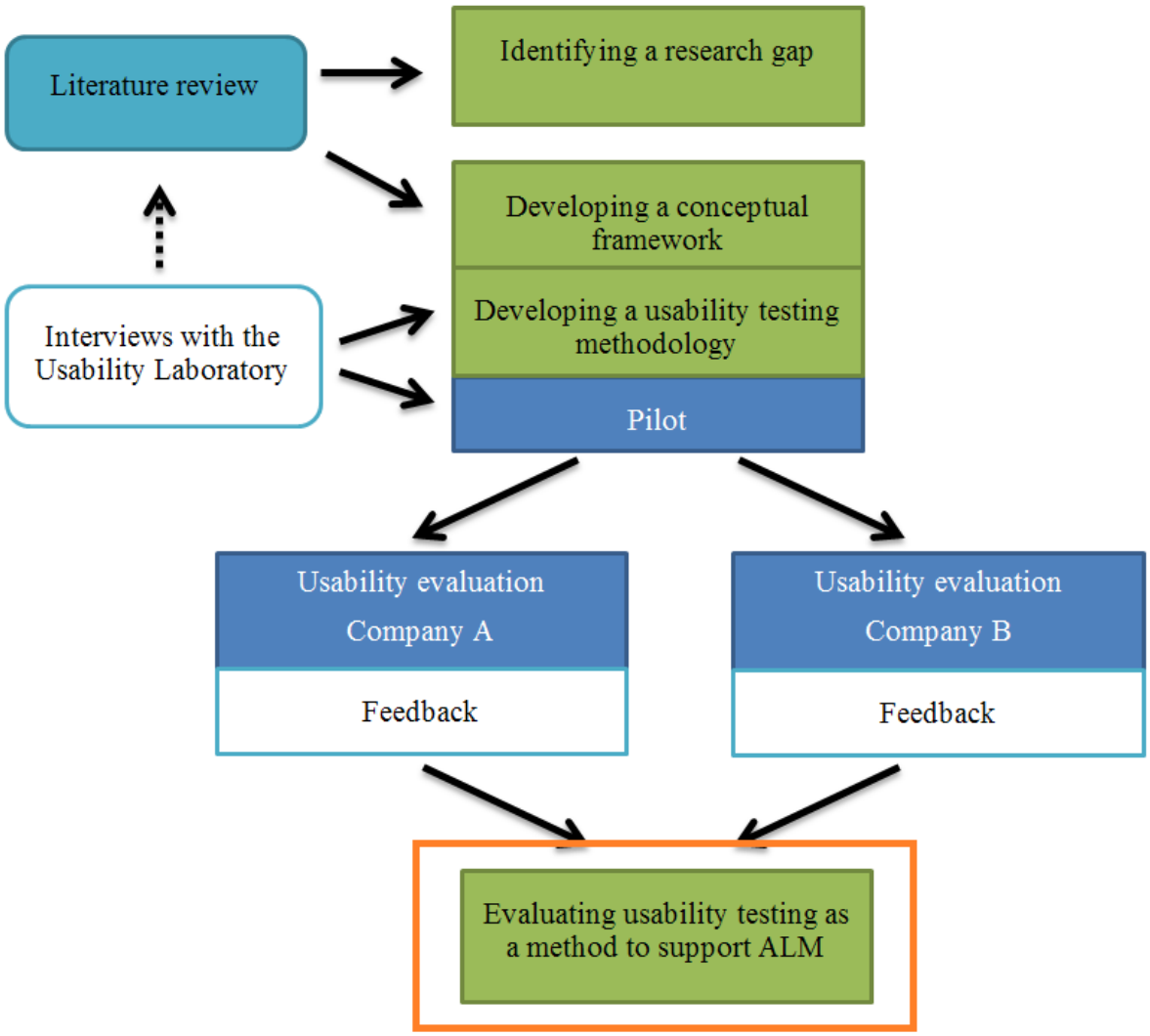
10. Could you see usability testing to be used continuously during the product lifecycle as a tool for continuous improvement?

The system specialist and project manager found this as a good idea. Once a year was stated as a good frequency of testing. The system specialist thought that usability evaluations are "good for prioritizing backlog". The company representatives stated, that if usability testing was done continuously, there is a need to change test users between rounds.

11. Would you yourself have time and/or willingness to conduct the usability evaluations, if the methodology was well documented?

"As a project manager no" (Project Manager, Company B). After discussing, the company representatives agreed that usability-testing skills could be useful in the organization. The usability testing training should go to the named owners of different business systems. It is unclear, however, how willing the business system owners would be to conduct usability testing.

5. ANALYSIS AND DISCUSSION OF THE RESULTS



5.1. Evaluating the usability testing methodology and format of the report

My first research question was to find out what kind of usability testing methodology could be used to evaluate IS throughout the application lifecycle. Based on prior research and expert interviews, I combined a usability testing methodology that would be easy enough to administer and produce data on not only user satisfaction but also user performance. SUS questionnaire measures all three aspects of usability. However, because of SUS's subjective nature, it could be

argued that it best reflect user satisfaction. Better data about user performance is gained by observing the user performing tasks and identifying common errors caused by the application.

The data from usability testing was presented in a form of usability evaluation. The overall reception of the format of the evaluations was positive. Especially well received was the positive feedback about system performance. It is important to note, that even though usability testing aims at finding opportunities for improvement, for management purposes it is also important to identify successes. Use of illustrations in the report was complimented.

Comparison of systems before and after updates got a mixed reception. Company A appreciated benchmarking to the old system, whereas company B felt, that it is better to focus only on the new system. If results from usability testing are used to further develop an application, the comparison is not necessary. However, if the data would be used to evaluating, for example a project, comparison should be made.

Comparison to wider data from Bangor et al. 2008 (Figure 7) was not found especially useful, as the aggregate data could come from very different products. In order for the comparison to be meaningful, it should be made to similar products. The graph by Kortum and Bangor, 2013 (Figure 8) gives benchmarking points to both commercial and consumer products.

The usability tests were conducted in Finnish and translation of SUS by Työterveyslaitos, (no year) was used. The translation uses the word “tuote” (product) to describe what is being evaluated. In the original SUS questionnaire by Brooke (1996) the word “system” is used. When conducting the usability tests, several test users mentioned that the word “product” seems weird when answering questions about an application. In this case it didn’t matter, as I could tell that the questionnaire is about the system in question. However, if SUS questionnaire is used without an instructor, the subject of the questionnaire should be clearly stated in the questions. Instead of using the Finnish translation of “tuote”, it would be better to use a more descriptive word that fits the tested application or systems. Such could be for example “järjestelmä” or the name of the system. Usability testing can be used to evaluate tangible goods, where the word “tuote” would be appropriate. Most important thing, if translating the original questions, is that the questionnaire is understood by the test users in the context of each individual study.

All in all, the developed usability testing methodology was accepted by the companies, and the format of presenting the results was well received. Because there were no major complaints about the method of collecting the usability data, we can evaluate usability testing's applicability for ALM. If there would have been heavy criticism towards the methodology, this would not be possible before fixing the methodology and repeating the usability evaluations.

5.2. Evaluating usability testing as an ALM tool

The second research question was: "Will company representatives see the developed usability testing methodology as a useful tool for ALM at different phases and different aspects of the application lifecycle?" In my conceptual framework, I suggested that usability data should be used to improve decision making on the governance aspect of ALM. Data would be collected from the operations and development aspects, would be used by the governance aspect, and the more informed decisions would result to benefits for development and operations. For development this would be seen in better project success rate and for operations it would mean more usable software. Usable here means both user satisfaction, but also more efficient and effective usage, as defined by the ISO 9241-171standard. This would lead to net benefits for the whole organization, as predicted by DeLone and McLean (2002).

5.2.1. Usability data for the strategic management level

In part 2.6. I suggested dividing the governance aspect of ALM (from Chappell, 2008) into strategic- and project management levels. On strategic level, the usability data would be used to project portfolio management and application portfolio management. Data would be used to evaluate projects and applications. Vendor management falls in between of project management and strategic management and could be considered as part of both. Vendor selection could be seen as a strategic management task, but vendor relations during the project are a part of project management. In this discussion, I will treat vendor management as part of strategic management.

Project manager of Company B stated how "specialists are not always very skillful in specifying requirements". This means that people who know what needs to be achieved are not always skilled in translating these needs into software requirements or functionalities. This is precisely

something that I assumed that the vendor could be able to help with, if there was an incentive to do so. I assumed that such incentive could be a contractual clause, where the vendor commits to certain level of usability, measured for example with an improvement in the SUS score of the new system compared to the old one. However, neither one of the companies found it especially appealing to have usability metrics written in contracts with vendors, or as a goal for internal project organizations. Exception was mentioned by Company B representatives. When something completely new was created, overall usability performance could maybe be used as an additional goal. Usability metrics for contractual clauses seems to have very limited use for ALM. The interviews suggest that only in the initial development (early stage of the lifecycle) could usability metrics be used as goals on contractual level.

Based on the interviews, it seems that perceived subjectivity of both quantitative and qualitative usability metrics make them poor requirements to be written in contracts or project goals that might trigger bonuses or sanctions. Perceived subjectivity by the companies means the same thing as weak reliability of the results (as explained by Nielsen, 1993). The companies felt, that a small sample cannot grasp the true quality of the system, and gives too much emphasis on the opinion of a small group of users. The companies don't feel that small samples are as descriptive as the experts from Strategic Usability Group suggested. It might also be that even if companies trusted the reliability of the results, possible vendors might not accept the results based on a small sample, if the results were not favourable for the vendor.

In addition to subjectivity, Company B representatives raised the dilemma of possible conflicting goals. The company representatives from both Company A and Company B stressed that the projects must deliver functional products. It is not enough that a product is usable if it, for example, has security issues. Tradeoffs between security and usability are common. A good example about this is passwords. Passwords are not very usable, but we still prefer to have passwords in most of the ICT systems we use. If overall usability was heavily stressed in project goals, it might encourage to compromise on some technical aspects (such as security) that are still critical for project success. It is also not always possible to have both strict usability and security goals, because they might be mutually exclusive. Even if achieving both should be possible, it might require some new innovations and creativity, that might not be possible on the

given budget and schedule. Having impossible goals is very de-motivating and should be avoided. Possible conflicting goals that can arise with heavy focus on usability is thus another reason, in addition to perceived subjectivity of results, for not to use explicit usability goals in contracts.

The interviews showed that usability might still have its uses in the governance aspect of ALM. An important reason for Company A to conduct usability testing was to evaluate the work of a vendor. Both companies stated that they usually have, and prefer to have, long lasting relationships with their vendors. Having some kind of measures of system development performance over long term relationships can help organizations to better evaluate their vendor relationships and find areas that need more attention in the future. For vendor management, usability thus seems to be unsuitable metrics to be written on contracts, but could be used to evaluate vendors on a general level over the long lasting relationships.

For project portfolio management, strategic managers need to evaluate and compare different project successes. The interviews revealed a perceived tradeoff between usability and the cost of a system. This is especially true, if usability is improved outside the initial development. Developers can make quite usable programs, but making changes can be very expensive. Usability issues and suggestions are difficult to communicate in terms of cost and benefits. An improvement to the usability evaluation report, as articulated by representatives from Company B, would be to assign cost to each usability issue, as well as price of fixing it. This would make usability testing much more valuable as a management tool. However, user satisfaction can be a tricky variable to put a value on. One concrete way to do this would be to measure time savings of the user that could be achieved by improving usability of system.

There is also the same problem with guiding internal developers with usability goals than with the vendor management. Usability might be conflicting with important technical requirements, which could make usability a demotivating goal.

Application portfolio management requires having an understanding of a company's application landscape. Simon et al. (2009) stresses the need of a permanent approach to measure applications over time. Both companies felt that it would be a good idea to measure the usability of a system

on regular intervals. A suitable timeframe, according to both of the companies, would be once a year.

For strategic management level of the governance aspect of ALM, usability testing would be most useful for long-term evaluation of applications and vendors. Use of usability goals in contracts was not embraced by the company representatives. Reason for this was the perceived subjectivity of usability testing, as well as the perceived possible conflicts between usability and technical functions as well as usability and cost.

Company B stressed the importance of cost for decision makers. Simon et al. (2010) stress that decisions in APM need to rely heavily on cost-benefit analysis. Company B wished usability evaluations that articulate a cost of the individual usability issues. Cost could be used to justify fixing possible problems. Lack of the cost dimension is a problem for usability testing. We could make estimates of cost by measuring time lost in errors for certain tasks, but the figures would be estimates as best. This is why usability evaluations can only be part of the data that strategic managers need for successful ALM.

5.2.2. Usability data for the project management level

In the suggested framework, project management level of the governance aspect deals with defining and adjusting project goals during application creation or modification. The interviews confirmed that there are challenges in setting scopes in ICT development projects to some extent. Company A representatives stated that they have not experienced challenges, but did agree that it is useful to use end user input when planning functionality of an ICT project outcome. Company B representatives did state that they have experienced some challenges in setting project scopes. Both of the companies agreed, that it is valuable to get input from the real users for the development of applications. Company B representatives told, how getting end user input lessens the amount of company politics in the planning phases, as it is rare for professionals to go consciously against the need of the (internal or external) customer.

Based on the interviews, usability testing is most appreciated for the project management level of ALM. This is because usability testing gives the best payoff during development. On the

strategic management aspect, usability testing data can be accessed when it is, in a sense, already too late. If there are any problems after the project phase, fixing them will be costly. However, improving the design in middle of the development via prototypes will be much cheaper. The organization already has a project team in place to make the changes, and changes in the development phase can be done without disturbing the operations. There is a risk that iterative development will be slightly more expensive than traditional waterfall model, but the total lifecycle cost should be lower, as fewer changes are needed in the later phases of the application lifecycle.

5.2.3. Usability data for the operations aspect

In my initial model, I illustrated how usability testing creates a flow of data from operations and development for the decision makers at governance level. However, my data collection showed that the flow of data goes two ways. Both Company A and Company B representatives were pointed out the possible effect usability testing might have for the user. The main difference between the two company projects was, that Company A had external and Company B internal users. Still both were concerned about the effect of usability testing on the users. The interviewees told, that with both internal and external users, asking feedback too often can be dangerous. Especially if the pace of changes that can be made is slow. When conducting usability testing, it is important to manage the expectations of the test users after the sessions. It can be stated, that some changes might not be technically possible and sometimes compromises have to be made. A long enough period between testing (and changes) should also be given, as users need time to accustom to changes. It is likely, that the effects on business are stronger if test users are real clients. However, we should not neglect the importance of the employee happiness. They key lesson is, that organizations should not engage in usability testing with real users, if there is no resources or intention to make any changes in the future. Asking for feedback and doing nothing is an easy way to frustrate users. This is a possible risk of doing regular usability testing just to collect data, without having a planned development project in mind.

Usability testing can also have a positive effect on the test users. Customers of Company A, who acted as test users, appreciated how the company is collecting user feedback and really putting

effort into satisfying customer needs. Users from Company B thought that usability testing was a great way to learn about the new system. They thought that having time, without customer interruptions, to "play around" with the new system made them more likely to use it in the future. Lassila and Brancheau (1997) argued that the amount of use is greatly dependent on the training and support users get when introduced with a new system. Usability testing can in a way, be part of the user training. It also sends a message, that the employer wants to provide tools, which the employees are satisfied with. Usability testing is thus more than just collecting data. It also sends a message to the end users. The way the data is then used by decision makers determines whether the message is considered positive or negative by the users.

5.2.4. Choosing the sample for usability testing

Controlling the sample for usability testing was the key technique I suggested to improve the reliability of results. Based on prior research, I suggested within-subjects testing (Nielsen, 1993) as well as heterogeneous samples in terms of age, gender and experience (explaining demographic factors for technology acceptance by Venkatesh et al., 2003). Controlling the sample proved to be the aspect of usability testing where my suggested methodology and the realities of the real world were farthest apart. During the research I learned, that testers can't be choosers. In reality, we have very little control over who is going to be our test users. Especially Company A, who had external test users, saw this as a constraint of usability testing. It is problematic to ask for customer feedback too often, as well as to find customers who are willing to give enough time that is needed for proper usability testing. Having the same test users over time would be practically impossible, if the users were external.

Company B, which had internal test users, thought that test users should be rotated. This is to gain as wide feedback as possible, but also to not bother the same people too frequently. This means, that even if access to internal end users would be easier, companies might be reluctant to engage in within-subjects testing over the long-term. By testing systems before and after changes with the same users, we can still achieve within-subjects testing for a section of the application lifecycle.

Because of weak control over the sample, usability tasting's reliability over long-term, even with controlled methodology, is decreased.

5.3. Refined framework: Usability testing throughout the application lifecycle

Based on the interviews, I refined the initial framework to take into account the two way flow of data during usability testing. The new framework also takes into account, that users need time to use an application before evaluation.

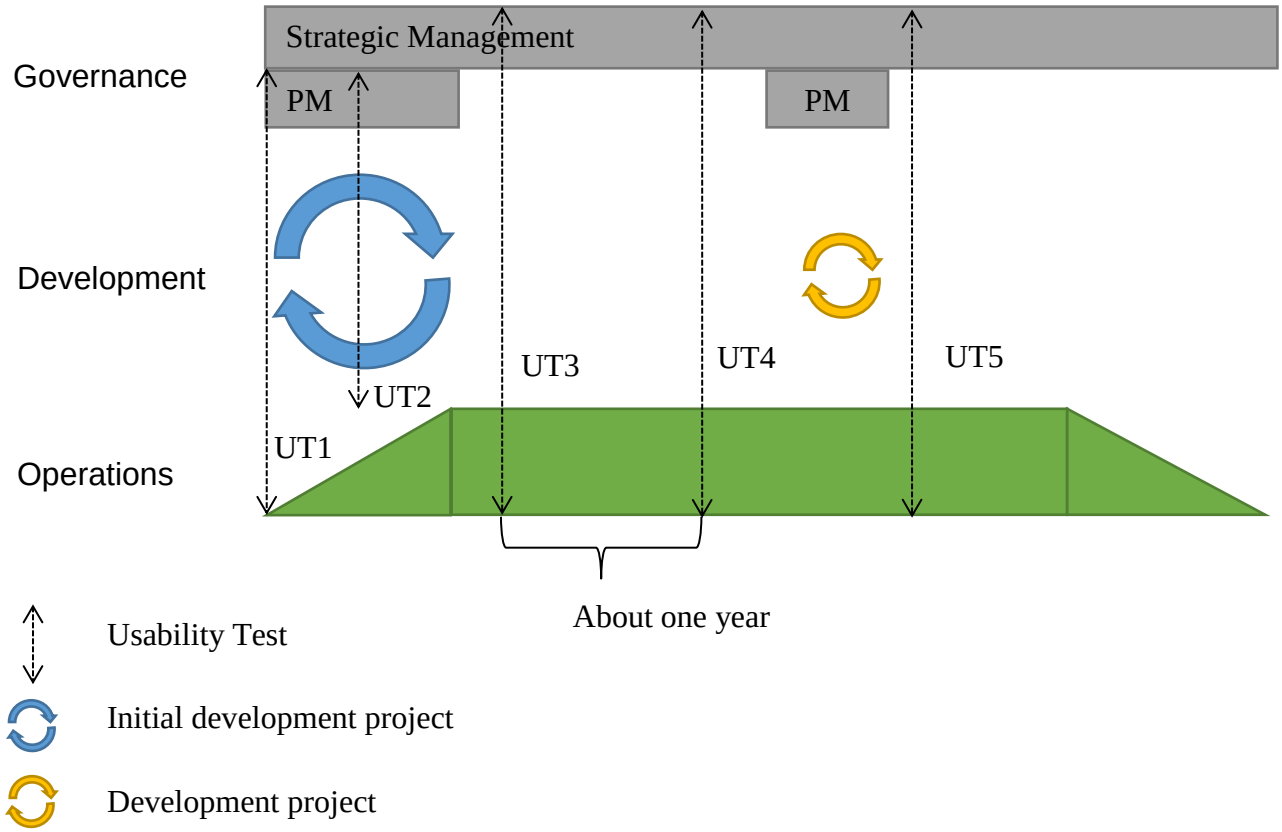


Figure 14: Usability testing throughout the application lifecycle

The figure above starts from the initial usability test (UT1) to help determine what functionalities should be added to the new system. In reality, UT1 could be the last usability test of the previous

system. UT2 shows a mid-development usability test to confirm that development is on the right track. Because the research showed that usability metrics would not be used by companies to cause immediate action (such as paying of bonuses) there is no need to test the application right after it is launched to production. Instead, some time (such as a month) can be given for users to get to accustomed with the system. After this a usability test of the system can be carried out in production (UT3 and UT5). Even if no changes are made, usability testing can be repeated after a year, to provide data for strategic management. In these mid-lifecycle tests, it is especially important to manage the expectations of the test users, and explain that there might not be any immediate changes.

My researched showed, that usability data would not be as widely used for strategic management as I suspected. Usability testing was seen, however, as very useful methodology for setting the scopes of development projects, as well as for prioritizing backlog after projects. For project success evaluation, there were some limitations. Based on the interviews, there seems to be three trade-offs, that are perceived to possibly conflict with usability goals for application development. These are **1) Usability/Cost trade-off**, **2) Usability/Functionality trade-off** and **3) Usability/Schedule trade-off**.

Even if usability testing would give best pay-offs at the development phase, there is no reason why the data shouldn't be shared with strategic managers, if it is anyways collected. A bank of data, collected with a uniform methodology over time, could be used to support the governance aspect of ALM. In addition to usability metrics, decision makers will pay heavy focus on cost in the governance of ALM.

In Table 1 I suggested that usability could be used to measure dependent factors of IS success identified by DeLone and McLean (2002). The dependent factors were net benefits to organization (which can also be independent factors, depending on the user) and user satisfaction. However, company representatives saw only limited potential for usability metrics as application success measure. Instead of measuring dependent variables, my research suggests, that the companies see usability testing primarily as a tool for improving the independent variables of IS success. Better designs will improve Information and System quality and have a positive effect

on the dependent success factors. Interaction with users during early development is also communication towards the users and could have a positive effect to the service quality independent factor. This requires that the user feedback results into actions, creating a sense of service towards the user.

D&M IS SM suggests to focus measurement of IS success to the dependent factors, that are mainly related to the end user. These are intention to use, user satisfaction, and net benefits resulting from user performance. However, the companies appreciate measures, such as security, which might not be consciously considered by the regular user. Security is a part of the System Quality independent factor in D&M IS SM. However, based on these two projects, I suspect that increase in system security might not lead to increase in user satisfaction and intention to use. Quite the opposite, complicated password procedures and automatic log-outs from a system might have a negative effect on the user satisfaction, even though the features are considered beneficial to the organization by managers of the governance aspect.

Johnson et al. (2007) discussed the limited use of usability testing in real business. The interviews confirmed that organizations did not have a lot of prior usability testing experience. However, the overall attitude towards usability testing was positive. The developed framework, illustrated in Figure 14, shows that usability testing can have a positive effect not only to the development, but also on the operational and governance aspects of ALM. This improves the cost-benefit ratio of mid-lifecycle usability testing, as the same data can be used by several actors in the organization.

Prior research of both ALM and usability testing concentrated on the early phases of ALM, namely the initial development. The companies shared the view that this is where usability testing has the best pay-off, and strongest positive effect. Better initial design can make the development a little bit slower and more costly, but is likely to reduce the overall lifetime cost of an application. However, the interviews showed that companies would be willing to engage in usability testing throughout the application lifecycle, also after the initial development.

5.4. Limitations of the developed framework

Application Lifecycle Management is a vast subject. I have tried to combine both the timeframe (from development to deletion) and the different organizational layers (operations, development and governance) with a very specific data collection method (usability testing). I'm afraid, that the end result has the same risk that was identified by Seddon (1997) about the IS success model by DeLone and McLean (1992): multiple meanings in the model might cause "slippage" from one concept to another. In the framework, the start and end of a single application lifecycle appear to be quite clear. Development is also illustrated to be done always in a project form. In reality, it can be challenging to decide which counts as a development of one application, and what as a creation of a totally new application with its own lifecycle. It is also likely, that some form of development goes on outside the formal projects. The division between different phases of the lifecycle can be challenging to identify in the real world. Other risk lies in the division of the different aspects. In reality, people in organizations can have multiple roles. It is thus likely that some people might actually carry on functions on several of the aspects of ALM.

The developed framework is suggested to be very universal. I have tested the framework with an application intended as an internal tool and with an application intended as external product. Intuition suggests that usability and user satisfaction would be more business critical for an external product than for an internal tool. However, Venkatesh and Davis (2000) argue that as user experience increases, the effect of subjective norm decreases. Based on this, I argued that usability could be as important for internal tools, as companies have little power to force employees to use technology based tools in the long-term. This was tested only with a limited sample with one company from both groups. In addition, both of the companies that participated in this research operated in the B2B market. It might be, that the suggested framework has different levels of validity for different types of organizations and businesses. Validation with a larger and more heterogeneous sample is needed.

There is also one major limitation in the actual usability testing methodology. The aim in developing the methodology was to make it straightforward enough, that organizations can conduct the usability evaluations internally. However, at least the managers and specialists, who

gave their feedback for the methodology, were not too excited about conducting the usability evaluations themselves. Usability testing was seen as an additional task to already busy schedules. Upon further thinking, there are also the ethical aspects of usability testing that need to be taken into account. As Nielsen (1993) stated, it is important that the test users can retain anonymity. This will allow users to act naturally with the applications as well as give honest, even critical feedback. If the conductors of usability testing are from the same organization the atmosphere of testing might not be relaxed and natural enough. For example, project managers might be in a supervising position to test user, or at least might potentially be in supervising positions in the future. This is likely to affect the willingness of the test users to show their uncertainty with an interface, or criticize the tested software.

5.5. Suggestions for future research

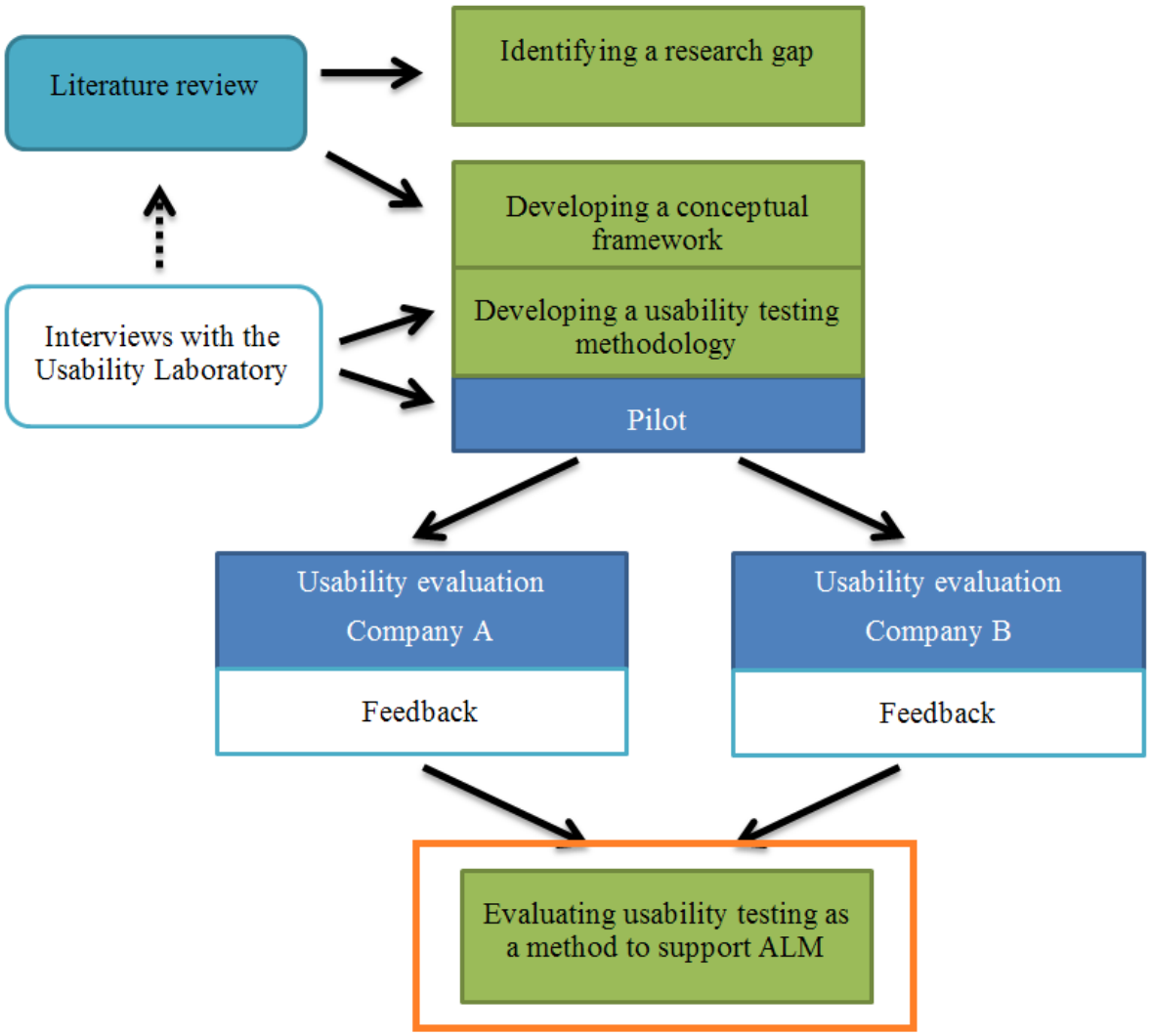
My research was limited to B2B businesses. There was also the assumption that usability is as critical for development of internal tools and external products. This assumption should be challenged by repeating the research with a larger sample of both types of organizations. Especially important would be to test the developed framework with a company that provides applications for the consumer market. Further research should be carried with software vendors for both B2B and B2C markets as well as organizations that do not produce applications as external products, but only as internal tools.

The results were somewhat surprising, as the utility of usability data for the strategic management level of ALM was seen more limited, than what I hypothesized with the initial framework. My data collection did focus on the project management level and the development aspect of ALM. It would be important to collect feedback about the framework also from managers who are involved in project portfolio management and vendor portfolio management, to see if they share the views of the project and development managers.

My research suggested that one of the main limitations of usability testing in real business is usability evaluations' lack of a cost dimension. Further development of usability testing

methodologies should focus on developing methods for measuring and articulating costs of possible usability issues. This is a tall order. The measurements of cost / benefit are likely to vary on many factors and difficult to derive from available usability testing methods. It is, however, something that would increase usability testing's utility for businesses.

6. CONCLUSIONS



Aim of this thesis was to address the lack of methods for evaluating applications in ALM, as well as the limited treatment of the later phases of application lifecycles in ALM literature. On other words, I attempted to suggest increase in available data for the governance aspect of ALM to support strategic decision-making. Based on prior research, usability testing seemed like a good tool to provide data from operations and development aspects of ALM to the governance aspect. I tested this assumption through two separate company projects. The research suggests that

usability testing can have utility beyond its design functions. However, engaging with users is always two-way communication, and organizations engaging in usability testing should be prepared to act on the collected data.

My first research question was to find out a suitable usability testing methodology for ALM. I suggested the use of SUS questionnaire via within-subjects testing, complimented with an array of other techniques. The suggested methodology increases the reliability of the widely used System Usability Scale even with small samples, when comparing different systems with each other. There are benefits for evaluating systems before and after changes with usability testing. This might not always be possible with incremental changes, as there might not be an old version to test in same sessions with the new version. However, with any major changes, it is likely that both the old and the new version of an application will co-exist for some time. In these cases within-subjects testing should be used. This way even the subjective SUS scores can be used more objectively in showing the nature of change between two versions of a system. Over long-term, within-subjects testing might not be possible, because companies feel it is problematic to get access to the same users time after time. This is especially true if the test users are also customers. There are improvement opportunities for the methodology, mainly introducing a cost of the identified issues to the usability evaluations. Nevertheless, positive feedback from the companies suggests that development of the methodology was successful.

The second research question asked if usability testing (represented by the developed testing methodology) could be a useful ALM tool for data collection and decision-making. My research showed that companies appreciate usability testing more as a development and design tool, which is its initial purpose. Initial development phase is where usability testing offers the best pay-off. It is more sensible to conduct usability testing when designing the application, rather than fixing problems later. However, if usability testing data is gathered, it can also be utilized on the governance level of ALM. Building a database of usability metrics of key systems over time will have several benefits for organizations. In the short-term, it is a good way to prioritize incremental changes without company politics. In the long-term, having a view on development of usability of different systems can help to evaluate which vendors or which applications are best meeting company needs over time. Usability testing can also strengthen organizations'

relationships with the users. As with any management tool, usability evaluations must be used as a part of the overall picture, not as the only source of data of system performance. The answer to the second research question thus seems to be that company representatives see the developed usability testing methodology as useful to some extent for ALM. The use of usability metrics as contractual goals for project organizations or outside vendors was rejected (except perhaps when something completely new is being created). Even though companies were willing to engage in usability testing throughout the application lifecycle, greatest utility was still perceived to be obtained from usability testing at the early phases of the lifecycle.

In order to make usability testing more appealing to companies, we must continue to develop usability testing methodologies. Our aim must be to find ways to objectify results even with limited samples, as it is not easy for companies to find or engage real test users. We must also try to develop ways to introduce cost as a measure of severity of a given usability issue.

Communicating with the users is important and there is potential for usability testing as a part of ALM. As ICT is becoming more and more prevailing in all aspects of life, having more usable applications has value on its own. Usable applications can make people more effective and efficient, but also more satisfied and happy. For external users, happy user is likely to be a happy customer. However, my research suggests that usability data could be used to not only improve designs, but also to support the strategic management of the complex application-, vendor- and project portfolios that many companies need to juggle with.

REFERENCES

Artto, K., Martinsuo, M. & Kujala, J. (2011) ”Project business”, pp. 22 -23, Available from: http://pbgroup.aalto.fi/en/the_book_and_the_glossary/project_business_2011.pdf [Accessed 21 March 2013].

Bagozzi, R. (2007) "The Legacy of the Technology Acceptance Model and a Proposal for a Paradigm Shift." *Journal Of The Association For Information Systems,* Vol. 8, No. 4, pp. 244-254.

Bangor, A., Kortum, P. & Miller, J. (2008) “An Empirical Evaluation of the System Usability Scale”, *International Journal Of Human-Computer Interaction*, Vol. 24, No. 6, pp. 574-594.

Bellenger, L. (2003) “Avoiding Scope Creep Protects the Bottom Line*”, ASHRAE Journal*, Vol. 45, No. 10, p. 58.

Brooke, J. (1996) “SUS - A quick and dirty usability scale”, *White Paper*, Available from: http://hell.meiert.org/core/pdf/sus.pdf [Accessed 6 January 2013].

Chappell, D. (2008) "What is Application Lifecycle Management?", *White Paper,* Available from: http://www.davidchappell.com/WhatIsALM--Chappell.pdf [Accessed 18 September 2013].

Davis, F. (1989) “Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology”, *MIS Quarterly*, Vol. 13, No. 3,pp. 319-340.

DeLone, W. & McLean, E. (1992) “Information Systems Success: The Quest for the Dependent Variable”, *Information Systems Research*, Vol. 3, No. 1, pp. 60-95.

DeLone, W. & McLean, E. (2002) "The DeLone and McLean Model of Information Systems Success: A Ten-Year Update.", *Journal Of Management Information Systems* Vol. 19, No. 4, pp. 9-30.

Flinders, K. (2011) “Why big IT outsourcing projects fail”, *Computer Weekly*, 11.1.2011, pp.11-13.

Green, D. & Pearson, J. (2006), “Development of a web site usability instrument based on ISO 9241-11”, *Journal Of Computer Information Systems,* Vol. 47, No. 1, pp. 66-72.

Haydaya, P. & Marchildon, P. (2012), “Understanding product lifecycle management and supporting systems*”, Industrial Management & Data Systems*, Vol. 112, No. 4, pp. 559-583.

International Organization for Standardization (2013) “ISO 9241-171:2008”, *ISO Home Page,* Available from: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=39080 [Accessed 22 March 2013].

Johnson, R., Salvo, M. & Zoetewey, M. (2007) ”User-Centered Technology in Participatory Culture: Two Decades “Beyond a Narrow Conception of Usability Testing””, *IEEE Transactions on professional communication,* Vol. 50, No. 4, pp. 320- 331.

Kortum, P. & Bangor A. (2013) “Usability Ratings for Everyday Products Measured With the System Usability Scale” *International Journal of Human-Computer Interaction,* Vol. 19, No. 2, pp. 67 -76.

Kääriäinen, J. (2011) “Towards an Application Lifecycle Management Framework, *Dissertation*, *VTT Publications*, 759, pp. 1- 103. Available from: http://www.vtt.fi/inf/pdf/publications/2011/P759.pdf [Accessed 17.9.2013].

Kääriäinen, J. & Välimäki, A. (2011) “Get a grip on your distributed software development with application lifecycle management”, *International Journal Of Computer Applications in Technology,* Vol. 40, No. 3, pp. 181-190.

Lassila, K. & James, B. (1999) "Adoption and Utilization of Commercial Software Packages: Exploring Utilization Equilibria, Transitions, Triggers, and Tracks." *Journal Of Management Information Systems*, Vol. 16, No. 2, pp. 63-90.

Lee, J., Keil, M. & Kasi, V. (2012) ”The Effect of an Initial Budget and Schedule Goal on Software Project Escalation”, *Journal Of Management Information Systems*, Vol. 29, No. 1,pp. 53-78.

Lewis, J. & Sauro, J. (2009) “The factor structure of the system usability scale” *Human Centered Design, Lecture Notes in Computer Science*, 5619, pp. 94-103

Li, L. & Yetton, P. (2009) “Sponsorship and IT vendor management of projects*”, Journal Of Information Technology*, Vol. 24, No. 1, pp. 46-54.

Nielsen, J. (1993) Usability Engineering, Academic Press, San Diego, pp. 165-166, 169, 178 – 179, 184.

Rachuri, S., Subrahmanian, E., Bouras, A., Fenves, S., Foufou, S. & Ram, S. (2008) ”Information sharing and exchange in the context of product lifecycle management: Role of standards”, *Computer-Aided Design*, Vol. 40, pp. 789-800.

Rai, A., Lang, S. & Welker, R. (2002) “Assessing the Validity of IS Success Models: An Empirical Test and Theoretical Analysis”, *Information Systems Research*, Vol. 13, No. 1, pp. 50-69.

Riihiaho, S. (2000) “Experiences with usability evaluation methods” *Licentiate's thesis. Helsinki, Finland: Helsinki University of Technology* [unpublished], pp. 1-113. Available from: http://www.soberit.hut.fi/~sri/Riihiaho_thesis.pdf [Accessed 21 November 2012].

Rubin, J. & Chisnell, D. (2008) Handbook of Usability Testing: How to Plan, Design and Conduct Effective Tests, Wiley Publishing Inc., Indiapolis, p. 28.

Seddon, P. (1997) "A Respecification and Extension of the DeLone and McLean Model of IS Success", *Information Systems Research*, Vol. 8, No. 3, pp. 240-253.

Simon, D., Fischbach, K. & Schoder, D. (2010) "Application Portfolio Management - An Integrated Framework and a Software Tool Evaluation Approach", *Communications Of The Association For Information Systems*, Vol. 26, pp. 35-56.

Stark, J. (2011) Product Lifecycle Management, Springer, London, pp. 1 -16.

Vasiljeva, T. (2012) "Outsourcing of Software Development: Challenges For Management" *Journal of Business Management*, No. 6, pp. 40-46.

Venkatesh, V., Morris, M., Davis, B. & Davis, F. (2003) "User Acceptance of Information Technology: Toward a Unified View", *MIS Quarterly*, Vol. 27, No. 3, pp. 425-478.

Venkatesh, V. & Davis, F. (2000) "A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies", *Management Science*, Vol. 46, No. 2, pp. 186- 204.

Technopedia (2013), "Application Lifecycle Management (ALM)". Available from: http://www.techopedia.com/definition/4431/application-lifecycle-management-alm [Accessed 25th October 2012]

Työterveyslaitos (no year) "Käytettävyydellä potkua tuotekehitykseen", p.21. Available from: http://www.ttl.fi/fi/ergonomia/menetelmat/Documents/kaytettavyydella_potkua_tuotekehityksseen.pdf [Accessed 25th October 2012].

Wepopedia (2013) "Application Lifecycle Management". Available from: http://www.webopedia.com/TERM/A/Application_Lifecycle_Management.html [Accessed 2nd November 2013]

Yuthas, K. & Young, S. (1988) "Material matters: Assessing the effectiveness of materials management IS", *Information & Management*, Vol. 33, No. 3, pp. 115-124.

INTERVIEWS

Development Manager, Company A, Espoo, 26.11.2012.

Project Manager and Development Manager, Company A, Helsinki, 24.5.2013.

Project Manager and System Specialist, Company B, Helsinki, 3.5.2013.

Riihiaho, S., Rämanen, J., Kaipio, J., Haanperä, T., Strategic Usability Group in Aalto University, Espoo, 6.11.2012.

Riihiaho, S., Rämanen, J., Kaipio, J., Haanperä, T., Strategic Usability Group in Aalto University, Espoo, 20.11.2012.

Riihiaho, S., Rämanen, J., Kaipio, J., Haanperä, T., Strategic Usability Group in Aalto University, Espoo, 26.11.2012.

Riihiaho, S., Rämanen, J., Kaipio, J., Haanperä, T., Strategic Usability Group in Aalto University, Espoo, 7.1.2013.

APPENDICES

Appendix 1: System Usability Scale in Finnish

System Usability Scale

Testaamani tuote:_____

	Täysin eri mieltä				Täysin samaa mieltä
1. Olen sitä mieltä, että voisin käyttää tätä tuotetta säännöllisesti.	1	2	3	4	5
2. Tuote on mielestäni liian monimutkainen.	1	2	3	4	5
3. Tuotetta on mielestäni helppo käyttää.	1	2	3	4	5
4. Mielestäni tuotteen käytön oppiminen vaatii kokeneen käyttäjän opastusta.	1	2	3	4	5
5. Mielestäni tuotteen eri toiminnot ovat liitetty toisiinsa onnistuneesti.	1	2	3	4	5
6. Mielestäni tuotteessa on liikaa epäjohdonmukaisuuksia.	1	2	3	4	5
7. Uskon, että useimmat oppivat käyttämään tuotetta hyvin nopeasti.	1	2	3	4	5
8. Mielestäni tuote on hyvin kömpelö käyttää.	1	2	3	4	5
9. Tunsin oloni hyvin luottavaiseksi tuotetta käyttäessäni.	1	2	3	4	5
10. Mielestäni ennen tuotteen käyttöä pitää opetella paljon uusia asioita	1	2	3	4	5

Työterveyslaitos (no year) *Käytettävyydellä potkua tuotekehitykseen* p.21

Appendix 2: Background information questionnaire

Kiitos, että osallistut [Järjestelmän nimi] käytettävyystestiin. Kaikki vastaukset käsitellään luottamuksellisesti.

Vastaa ensin alla oleviin taustakysymyksiin.

Ikä:

Alle 20	20 – 30	31 -40	41 - 50	51 – 60	yli 60

Työnkuva / Työnimike:

Kuinka kauan olet käyttänyt [vanhan järjestelmän nimi] (mikäli alle vuoden, merkkaa kuukaudet)

	Vuotta / kk (yliviivaa turha)

Kuinka montaa eri päätoiminnallisuutta käytät työssäsi? Kpl

Kuinka usein päivässä käytät [vanhan järjestelmän nimi]? n. kertaa päivässä

Kuinka usein käytät Internetiä vapaa-ajallasi

päivittäin	4-6 päivänä viikossa	3-4 päivänä viikossa	1-2 päivänä viikossa	En juuri käytä internettiä vapaa-ajalla

Millä laitteella käytät Internetiä vapaa-ajallasi (voit raksia usean vaihtoehdon)

Pöytäkone	
Kannettava tietokone	
Tabletti	
Matkapuhelin	

Appendix 3: SUS for the old system

Alla on väittämiä **nykyisestä** käyttöliittymästä. Vastaa väittämiin raksimalla rasti ruutuun.

1 = vahvasti eri mieltä
3 = ei samaa eikä eri mieltä
5 = vahvasti samaa mieltä

	eri mieltä				**samaa mieltä**
	1	2	3	4	5
1. Olen sitä mieltä, että voisin käyttää tätä tuotetta säännöllisesti					
2. Tuote on mielestäni liian monimutkainen					
3. Tuotetta on mielestäni helppo käyttää					
4. Mielestäni tuotteen käytön oppiminen vaatii kokeneen käyttäjän opastusta					
5. Mielestäni tuotteen eri toiminnot ovat liitetty toisiinsa onnistuneesti					
6. Mielestäni tuotteessa on liikaa epäjohdonmukaisuuksia					
7. Uskon, että useimmat oppivat käyttämään tuotetta hyvin nopeasti					
8. Mielestäni tuote on hyvin kömpelö käyttää					
9. Tunsin oloni hyvin luottavaiseksi tuotetta käyttäessäni					
10. Mielestäni ennen tuotteen käyttöä pitää opetella paljon uusia asioita					

Appendix 4: SUS for the new system

Alla on väittämiä **uudesta** käyttöliittymästä. Vastaa väittämiin raksimalla rasti ruutuun.

1 = vahvasti eri mieltä
3 = ei samaa eikä eri mieltä
5 = vahvasti samaa mieltä

	eri mieltä				**samaa mieltä**
	1	2	3	4	5
1. Olen sitä mieltä, että voisin käyttää tätä tuotetta säännöllisesti					
2. Tuote on mielestäni liian monimutkainen					
3. Tuotetta on mielestäni helppo käyttää					
4. Mielestäni tuotteen käytön oppiminen vaatii kokeneen käyttäjän opastusta					
5. Mielestäni tuotteen eri toiminnot ovat liitetty toisiinsa onnistuneesti					
6. Mielestäni tuotteessa on liikaa epäjohdonmukaisuuksia					
7. Uskon, että useimmat oppivat käyttämään tuotetta hyvin nopeasti					
8. Mielestäni tuote on hyvin kömpelö käyttää					
9. Tunsin oloni hyvin luottavaiseksi tuotetta käyttäessäni					
10. Mielestäni ennen tuotteen käyttöä pitää opetella paljon uusia asioita					