Aalto University
School of Science
Degree Programme of Computer Science and Engineering

Antti Paananen

Improving In-House Software Development Process:

A User-Centered Approach

Master's Thesis
Espoo, June 5, 2013

DRAFT! — October 15, 2013 — DRAFT!

Supervisor: Professor Marko Nieminen
Instructor: Jouni Kuusinen M.Sc. (Tech.)

Aalto University
School of Science
Degree Programme of Computer Science and Engineering

ABSTRACT OF
MASTER'S THESIS

Author:	Antti Paananen		
Title:			
Improving In-House Software Development Process: A User-Centered Approach			
Date:	June 5, 2013	**Pages:**	45
Professorship:	Usability and User Interfaces	**Code:**	T-121
Supervisor:	Professor Marko Nieminen		
Instructor:	Jouni Kuusinen M.Sc. (Tech.)		
-			
Keywords:	Usability, ERP, Software development process, Process measurement, Cognitive Walkthrough, Remote usability evaluation, Automatic usability logging, SUS, Contextual Inquiry, ISI, Software process improvement		
Language:	English		

Aalto-yliopisto
Perustieteiden korkeakoulu
Tietotekniikan tutkinto-ohjelma

DIPLOMITYÖN TIIVISTELMÄ

Tekijä:	Antti Paananen		
Työn nimi: Yrityksen sisäisen ohjelmistokehitysprosessin parantaminen: Käyttäjäkeskeinen lähestymistapa			
Päiväys:	5. kesäkuuta 2013	**Sivumäärä:**	45
Professuuri:	Käytettävyys ja käyttöliittymät	**Koodi:**	T-121
Valvoja:	Professori Marko Nieminen		
Ohjaaja:	Filosofian maisteri Jouni Kuusinen		
-			
Asiasanat:	Käytettävyys, ERP, Ohjelmistokehitysprosessi, Prosessimittaus, Kognitiivinen läpikäynti, Käytettävyyden etäarviointi, Automaattinen käytettävyyslokitus, SUS, Kontekstuaalinen tutkimus, ISI, Ohjelmistoprosessin kehitys		
Kieli:	Englanti		

Acknowledgements

-

-

Espoo, June 5, 2013

Antti Paananen

Abbreviations

CD	Contextual Design
CI	Contextual Inquiry
CMMI	Capability Maturity Model Integration
CW	Cognitive Walkthrough
ERP	Enterprise Resource Planning
HCD	Human-centered design
HCI	Human-computer interaction
ISI	Interaction Sequence Illustration
IT	Information technology
SUS	System Usability Scale
UI	User interface
UMM	Usability Maturity Model
UX	User experience
SPI	Software process improvement

Contents

Chapter 1

Introduction

This chapter describes the background and reasoning for the thesis as well as the focus and the limitations of the study. This section also presents the research problems and the overall structure of this thesis.

1.1 Motivation and aim of the thesis

In the 1980s, when the usage of personal computers (PCs) became more common, software design practices were still falsely assuming that the users were knowledgeable and competent in computer science. As an outcome, a significant part of the users were practically incapable of using operating systems and applications.During these times, the concepts of Human Computer Interaction (HCI) and usability became important. Since then, the design processes of interactive software for common people has emphasized usability. This process or approach is called human-centered design (HCD). [7]

The term Enterprise Resource Planning (ERP) was invented in the early 1990s.[15] The purpose of the ERP software is to offer techniques and concepts for integrated and thorough management of business, as well as making it more efficient. The usage of ERP software has increased globally and nowadays even service organizations have invested a lot of resources in ERP implementation.[16, 5]

Despite the importance of the efficiency aspect, the usability of ERP systems is not a widely researched subject area. However, weaknesses in usability may lead into low productivity and make it harder for users to achieve their goals.[25]

The aim of this thesis is to examine how the usability of a service-oriented ERP system can be enhanced by integrating usability inquiries, inspections and measures into the software development processes.

This study examines a customer service related business process employed in the subscriber company, (described in more detail in section 1.2) and evaluates the state of its usability by using a variety of applicable methods:

- Contextual Inquiry to define the business process.
- Cognitive Walkthrough for usability inspection.
- Interaction Sequence Illustration (ISI) to measure the amount of interaction steps and to understand them.
- System Usability Scale (SUS) to give a global view of subjective assessments of usability.
- Remote usability evaluation (by using log data) to evaluate usability in distributed locations.

According to the ISO standard of human-centered design for interactive systems [8], many benefits can be gained by using human-centered methods as a part of software development process. The productivity of an individual user can be increased together with the operational efficiency of an organization. Usable and useful systems also reduce training and help-desk costs, as well as stress and discomfort because they are easily understood by users. In other words, human-centered design improves the UX (User experience). [8]

The benefit of human-centered design (for software development process) is the increased total life cycle of a product and the likelihood of the project succeeding on time and within the budget. The human-centered approach also decreases the risk of software being rejected by the users, or failing to meet the requirements. [8]

1.2 Background and research questions

The subscriber of this thesis is a middle-sized company which is offering information services globally. The company is using in-house software development in order to create dynamic and bespoke software solutions and thereby aiming for commercial efficiency. It is not using any specific software development methodology, but operates in iterative manner. Because of the fast pace of growth, the company is willing to reform their current ERP system as well as the whole software development process. This thesis aims to give answers to the following research questions:

- ***How usability methods can help to identify critical disparities in the usage of a system?***

  Understanding the differences in the system usage between individuals can help to understand and deploy best practices throughout the organization and therefore improve efficiency.

- ***How much the efficiency of use can be improved by utilizing the results of usability evaluations?***

  It is important to find the most effective and usable user interface solutions and thus decrease the average time spent on tasks.

- ***What usability methods can be practically joined with the software development process of an ERP system?***

  Finding practical and efficient usability methods to be joined with the software development process can improve the quality of end product and also raise the maturity level of the development process.

1.3 Scope and structure of the thesis

This thesis covers a research study about the usability of a process, which is executed in the ERP system, and employed by the customer service department of the subscriber company. This thesis aims to join the models of software process improvement (SPI) and the human-human centered design to attach usability perspective into the software development of the subscriber company. It covers only the human-centered approach to the development process. The literature research consists of the SPI, two of its models and a few usability methods. Though the target of the research is ERP software, literature about Enterprise Resource Planning is not covered in the thesis.

The first actual chapter of the thesis elucidates the models of software process improvement. The second chapter describes the usability methods used in this thesis in order to improve the software process. Every usability method used in the research is discussed carefully. The third chapter introduces the process experiment. It covers the experiment steps and reflects the to the process of human-centered design. It also includes the implementation details of the study. The fourth chapter analyzes the data acquired from the process and the implementation process itself. The Last chapter discussed and concludes the research study and the whole thesis.

Chapter 2

Software process improvement maturity models

According to O'Regan [20] the software process improvement is "a program of activities designed to improve the performance and maturity of the organization's software processes and the results of such a program". In practice, the aim of SPI is to meet the business goals more efficiently and for example to improve the software quality. In other words, it aims for smarter work and better software in less time. Many process models or frameworks exist for software process improvement and one of them, the Capability Maturity Model Integration (CMMI), is presented in this chapter. Because of the usability approach of this thesis, also a model emphasizing human-centeredness is introduces. The user-centered process model for SPI is called Usability Maturity Model (UMM). [20]

2.1 Capability Maturity Model Integration

The Capability Maturity Model Intergration was developed in the early 1990s by the Software Engineering Institute. Its purpose is to define best practices for software processes in an organization and thereby improve their maturity. In the case of this thesis, the object of interest is the development version of the model, called CMMI for Development (CMMI-DEV). It provides a carefully defined road map and structured approach for the software process improvement and allows to set improvement goals and priorities. The CMMI consist of five maturity levels and each level includes a number of process areas. These process areas consist of set of goals, which need to be implemented by the defined practices. These practices specify what needs to be done. A maturity level is achieved when all the process areas of that maturity level have been implemented. [20]

After the CMMI is initialized (at first level), the focus at level two is on project management practices such as requirements management and project planning. Level three requires procedures and standards for engineering. For example design, coding and testing should be defined for effective risk management and decision analysis. Process performance must be achieved within the defined limits on the fourth level of the CMMI. The implementation of the level also requires using metrics and setting goals for the performance. The last level of the model requires a culture of continuous improvement in the company. The possible defects need to be identified and actions taken to prevent them to re-occur. Each of the levels and their improvements forms the basis of the next level in the Capability Majority Model. [20]

The level representation of the CMMI is described in Figure 2.1 including the levels and the CMMI process areas. Every process area consist of **specific** and **generic** goals and practices. The specific goals and practices are unique for each process area, and describes what needs to be done to perform the process. The specific practices connected to the specific goals describes the activities to achieve those goals. The generic goals and practices, on the other hand, are common for all the process areas in the CMMI level. The implementation of the generic practices institutionalizes the process, meaning that the process is documented, defined and understood, and that the process users are appropriately trained. The generic goals could be for example to have managed, defined and optimized processes in the organization. [20]

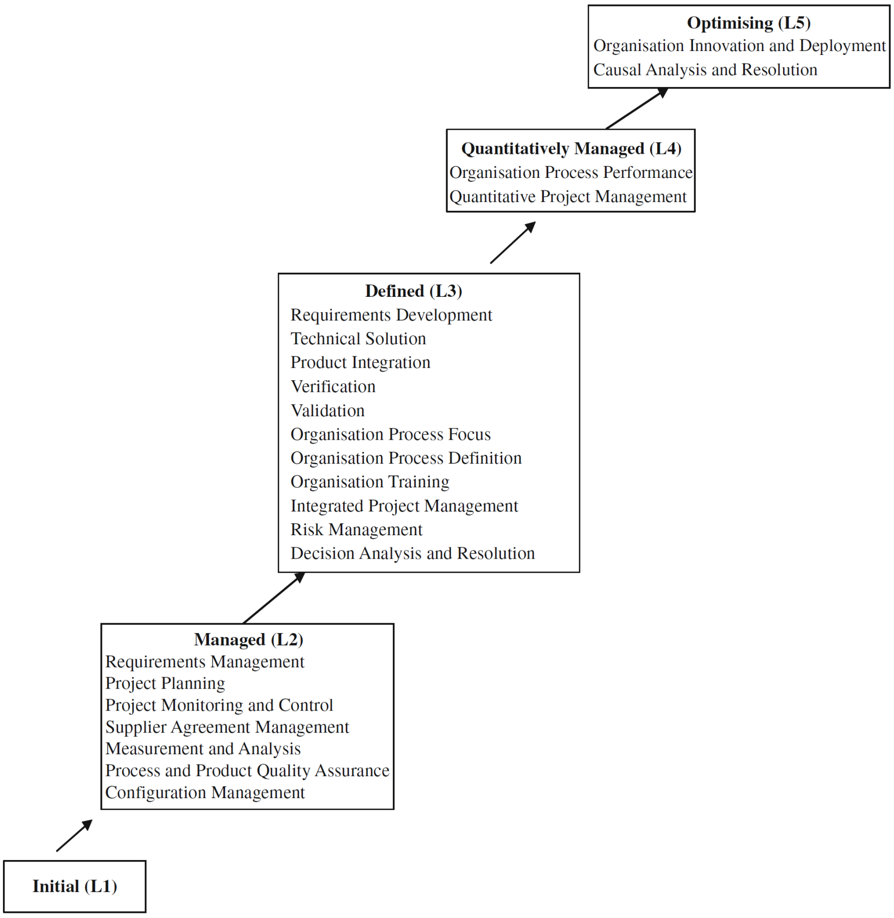

Figure 2.1: Capability Maturity Model Integration: Maturity levels. [20]

2.2 Usability Maturity Model

The Capability Maturity Model Integration is a good example of a model for software process improvement but it doesn't consider the human-centered part of system development. The Usability Maturity Model has been created as a scale to measure the human-centeredness of system development projects. In other words, UMM is a method to evaluate organization's capability to implement human-centered design. It has many corresponding elements with other SPI models, such as ISO TR 15504 standard, but it is still considered a stand-alone model. [9]

2.2.1 Maturity levels of the model

Like CMMI, also Usability Maturity Model consist of maturity levels (see Figure 2.2). In UMM there are six maturity levels and the first one of them is called level X, in which the need for human-centered activities are not recognized at all. In the level A the organization has recognized the need for improving systems quality of use. Level A1 is a problem recognition attribute and it describes the extent if understanding the problems. To achieve this level of maturity, the staff and the management need to be aware about the need of improvements. Level A2 describes the number of processes performed which provide input for human-centered activities. Information of user requirements should be included in the information collection and practices should be performed to gather this information. In level B, the organization indicates to the staff that the quality of use is considered as important attribute in the development and trains the staff to be aware that usability can be improved by considering the user requirements. The level B1 can be achieved by training the staff human-centered methods and principles of human-system interaction. The staff need focus on the users, for example by understanding the end users' skills, backgrounds and motivations. [9]

The human-centered processes are already implemented in the level C. Users are involved in specifying and testing the system and suitable human-centered techniques are being employed. Active involvement of users, the creation of UX, users defining the quality of use and continuous testing and feedback are required in order for the organization to achieve the level C1 maturity level. The level C2 on the other hand requires the use of appropriate usability methods, suitable facilities and tools, and maintaining usability techniques. Maintaining usability techniques includes reviewing the suitability of the methods and that the state-of-the art UI technologies are being used. The level C3 can be achieved by ensuring that the staff has the defined and required human-centered competences. In level D and in all the sublevels, the human-centered processes are integrated to the life cycle and quality processes of the system. Also, the required time and resources are targeted for these activities and interaction with other departments is successful, feedback process is administered and the design solutions are being iterated. [9]

The last level of UMM is level E and its sublevels, which requires the institutionalization of the human-centered processes, meaning that the organization gains benefit from its human-centeredness. On this maturity level, usability skills and engineering skills are used together and the usability defects are managed in a similar manner to other system defects. Also, human-centered process are being included in the projects and usability is systematically improved. Human-centredness has an influence to the whole organization. [9]

ID	Title
Level X	**Unrecognised**
	(no indicators)
Level A	**Recognised**
A1	Problem recognition attribute
A2	Performed processes attribute
Level B	**Considered**
B.1	Quality in use awareness attribute
B.2	User focus attribute
Level C	**Implemented**
C.1	User involvement attribute
C.2	Human factors technology attribute
C.3	Human factors skills attribute
Level D	**Integrated**
D.1	Integration attribute
D.2	Improvement attribute
D.3	Iteration attribute
Level E	**Institutionalised**
E.1	Human-centred leadership attribute
E.2	Organisational human-centredness attribute

Figure 2.2: Usability Maturity Model: Maturity levels and process attributes. [9]

2.2.2 Level transitions

The transitions between the maturity levels change the organization and these changes creates the basis for software process improvement. In the case of UMM, transitions represents also the improvement in usability consciousness.

The transition between maturity levels A and B is a cultural change, from experience based, to more user-centric engineering. In level A, the attitude against user-centeredness might be incredulous, but in level B the awareness about system being used by the people has been created. The transition from level B to C creates the cultural change of user being thought during the development. Also, the differences between analyst and end-users are being recognized. [9]

The Change from the level of considering the users to the implementation of the human-centered processes (from level C to D) requires the routine use of human factors expertise and human-centered methods and tools. The user involving the development process is considered normal in the level D. The transition from the level D to level E institutionalizes the human-centeredness. The system development is then embedded in a business driven culture, which changes the focus (of the development) from the functionality of the supporting systems to what the organization is able to do in general. In other words, the system functionality is not the core issue any longer. [9]

2.2.3 Utilization of the model

A member of staff need to be designated in charge of the quality of use. The first task of this employee is to examine the awareness of the human-centered principles within the organization by interviewing the managerial level. This information can then be used to assess the maturity level of the organization. However, more than one project should be assessed in order to gain wider understanding about the maturity of the organization. The performance assessment forms the basis of plans to review and improve the human-centered processes in the organization. [9]

Chapter 3

Methods for user-centered software process improvement

In order to be able to discover reliable research data, the research methods must be understood thoroughly. This study gathers data with a few types of usability methods which are selected according to their practicality and utility to the study context. In general, the methods used in this research study can be used as a part of human-centered design process. The reasoning for the choice of the methods is defined in the section 4.2.

3.1 Remote usability logging

There are many different ways to implement a remote usability evaluation. Typically it is accomplished by surveys or asking for feedback after the system has been deployed. This kind of data is important indicator of user satisfaction, but doesn't really give any specific details about the real system usage. However, the usage data is essential to isolate problems in usability. One approach to access the data remotely is to use data logging method, which is being discussed in this paper. In the context of usability,data logging means practices for mechanically recording the usage of a system. [13]

Usage data of an application can offer valuable information about users' actions and can therefore be utilized in the process of improving software's usability. Even if logging can not replace the traditional usability methods, it provides many advantages over them. Logging is automatic, objective and it doesn't require direct observation. The data is gathered from the actual running application. [3]

3.1.1 Evaluation process

According to Bateman et al. [3] log-based usability evaluation process consist typically of three stages (see Figure 3.1). The first stage is called *application instrumentation*. In application instrumentation the logging capability is added to the application. In other words, instrumentation is a process which determines what data will be logged from the usage of an application. In order to gain useful data, successful decisions in instrumentation stage are crucial. Bateman et al. assert that if wrong decisions are made and therefore large amount of low-level data has been collected, the data might be challenging to interpret and might not bring any value. On the other hand, they remark that if only high-level events are logged, internal structures may remain undiscovered. Consequently, both, low-level and high-level usage data need to be tracked and logged. Sometimes, when log data doesn't supply enough information for interpretation, contextual information is needed. Generally, it requires a significant amount of effort and vigilance to be able to gather all the essential data to be analyzed. [3]

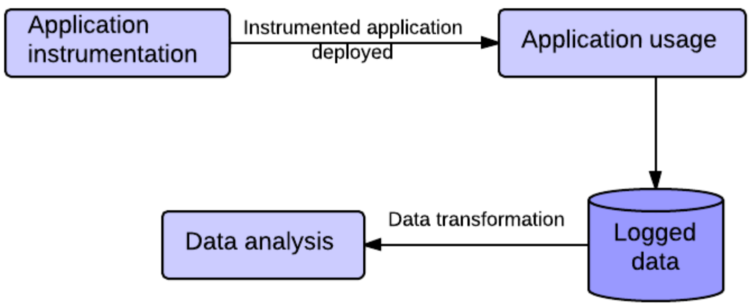

Figure 3.1: Process for log-based usability evaluation. [3]

The second stage of log-based usability evaluation process is *application usage*. It is a stage where the instrumented application is used in authentic or simulated situation. The log data is collected unobtrusively while the user is performing his or her tasks with the application. [3]

The final stage of the process is *data analysis*, in which a few different approaches can be used [3, 14]:

- Synchronize and searching
- Transforming event streams
- Performing counts and summary statistics

- Detecting, comparing and characterizing sequences
- Visualizing

This part of the process will be described in more detail in the following subsection Log data analysis.

3.1.2 Log data analysis

In order to gain beneficial usability information from user interface events, the log data has to be analyzed thoroughly. According to Hilbert and Redmiles [14], there are several approaches to sort out important usability information out of log data and to make it more understandable for humans. The first one of them is called ***synchronization and searching***.

It is often challenging to interpret user interface events alone and thereby discover valuable higher level events without any contextual information. The purpose of the synchronization and searching is to combine user interface event data with other informative data, such as observations or video recordings, in order to increase the understanding about the context of use. These two forms of data are complementary and can together provide wider understanding about the usability of a system. Observations or video recordings may supply additional information about the user interface events appeared in the log, or vice versa. However, there are few disadvantages with synchronization and searching techniques. For example video recording and observation typically require the presence of the observer. Furthermore, using video recording as a part of evaluation produces a lot of data and can be inefficient to analyze. It might also have an disturbing affect on user's behavior. [14]

The second approach is ***transformation***. Transformations combine selection, abstraction and recording to transform events into more beneficial form of information. This information can then be utilized for instance in detecting, comparing or characterizing human behavior patterns. Selection is basically segregating useful information out of mass of user interface related events by filtering out irrelevant data or by selecting the relevant data. Example of selection could be a situation where the user has been writing a lot of text (and thereby generated a lot of data in the log) and is trying to find save button from different menus. From the usability point of view, text inputs as log events are probably not that relevant, but time consuming browsing between different menus can be a sign from usability issues. [14]

Abstraction can be used to combine different events into a more understandable event sequences or patterns. For example, a user typing into different text fields and then clicking a button in a web page could be interpreted as a login activity. However, in this kind of situation user interface event logs must be supplemented with contextual data to be assured that the event sequence is really what it appears to be. Recording means generating new sets of events based on selection and abstraction. Less effort is required to analyze new sets of raw data because earlier analyzing techniques can be exploited. [14]

The third way of generating tangible usability information out of user interface log data is to use ***counts and summary statistics***. Counts and summary statistics are calculations based on usability-related metrics gathered from the log data. For instance, calculating the time spent on a specific task (performance time), can be critical usability information. [3, 14]

Detecting, comparing and characterizing sequences are approaches which utilize sequence information of events. Sequence detection refers to an action in which ready-defined sequences are tried to be identified from the mass of source sequences. Sequence comparison is executed by usability analyst and it is made between two sequences. These two sequences can be generated for example according to subject or subject groups. In any case, the purpose of sequence comparison is to compare actual usage against some predefined ideal usage. Sequence characterization uses the source sequences to create a model which summarizes all the features of interest in those source sequences. [3, 14]

The last approach for log-based data analysis is called ***visualization***. In visualization, transformed and analyzed data is presented in a graphical form. General way to visualize data is to use charts, but it is possible to use other visualization techniques such as heat maps based on clicks or mouse traveling. [3, 14]

3.2 Contextual Inquiry

Contextual Inquiry is a qualitative data-gathering and data-analysis methodology. It is adapted from psychology, anthropology and sociology. [21] In practice, Contextual Inquiry is an unstructured interview method, but it has some qualities which differs it from traditional interviews.[23] It was originally developed to meet three requirements. Firstly it was supposed to identify a design process for systems that will be used similarly in different business contexts and in different cultures. It was also supposed to identify a convenient process for gathering user information in limited time, and finally to identify a way to acquire information about users' work in eligible format. In addition to those requirements, the technique was noticed to be capable of much more. CI cherish participatory design and because of that quality, the users are able to involve in the design. Users contribute to the design by providing a deep understanding about the nature of the work. This is done through inquiry and it is used as a basis for fundamental work concepts. [24] According to Raven and Flanders [21] The Contextual Inquiry was developed in 1986 at Digital Equipment Corporation by human-computer interaction professionals. [21]

3.2.1 Fundamentals of Contextual Inquiry

Contextual Inquiry can be said to be an apprenticeship compressed in time. The basis of the method is premised on the idea of user being the expert instead of the interviewer, but unlike an apprentice, the interviewer neither learns the work by doing it, nor has the same amount of time available for learning. [4] CI differs from the traditional master-apprenticeship model in other ways also. A few fundamental principles of Contextual Inquiry are said to be essential in order to meet the specific needs of design problems [4, 1]. These principles are understanding the context of the work, creating a partnership, interpreting the work and steering the focus during the interview. [4, 21]

Understanding the **context** of the work is the baseline of Contextual Inquiry. To gain the understanding about the work structure, the interviewer must pursue understanding about the details of users' work and these details can be found by following the users' actions at work. In general, it is important that interviewer avoids gathering abstract or summarized information about the context.[4]

It is essential to create collaborative environment and a **partnership** between the user and the interviewer while the real life work structure and activities are tried to be understood thoroughly. Partnership is an equal relationship between the interviewer and the user. In comparison to traditional interview or master-apprentice approach, the partnership doesn't give any power advantage to either of the parties. Instead, it fosters the interviewer's expertise to see the work structure and the user's expertise to do the work. There are many advantages in partnership approach. For example by paying attention to the details and the structure of the work, interviewer can also teach the user to attend to them. In the best case scenario the interviewer and the user contemplates about the work structure and design possibilities together. In this kind of scenario it is common that the work is suspended, while the parties discuss about the work structure, and then return again. For interviewer asking feedback for design ideas is also encouraged.[4]

Even if a partnership should be created between the parties, the interviewer should still be able to steer the interview and keep the **focus** of the conversation on work-related topics relevant to the design [4]. The focus point should be decided before the research takes place, and data gathering should have a deliberated and precise goal. This goal or focus should depend on the information needs of the design.[1]

The success of Contextual Inquiry and system design depends on the facts gathered, but the facts are not enough. They are a starting point.*"From the fact, the observable event, the designers makes a hypothesis, an initial interpretation about what the fact means or the intent behind the fact."*[4] In other words, **interpretations** are needed and they are critical for the success of an inquiry. In the final version of the system interpretations have to be correct or the system fails. This is why it is important to share and validate the interpretations with the customer early enough. [4]

There are three types of requirements which should be considered in system development: Technical, business strategic and behavioral. The Contextual Inquiry is a part of Contextual Design (CD), which is a comprehensive design methodology. Contextual Design emerges within HCI and is used in requirement elicitation focusing on *behavioral requirements*. Therefore, CI can be also used as a practice for examining requirements of a system in authentic environment. [11, 19]

As soon as thorough understanding about the work is available, design for a system model can be created.[24]

3.2.2 Contextual Inquiry interviews

Practical preparations of CI includes careful planning. The first phase of planning is to set the focus for the research. Focus can be for example a definition of a problem which need to be solved. It creates ground rules for the interview and it is therefore easier for the interviewer to steer the conversation. After the focus has been set, the inquiry itself need to be designed. The challenge in inquiry design is to find a way to determine underlying issues which cause problems in the work. The approach for the design should be slightly different if the aim of the Contextual Inquiry is to support upgrading the system, creating a totally new system or redesigning the process. If CI is used to help the design process of a new system, a challenge is to get the designers and the users to work together in order to define new ways of working, and to develop a system design to support them. [4]

The structure of the interview is considerably straightforward. First task of the interviewer is to introduce the CI process and ask permission to record the conversation and the work. The interviewer has to also make it clear that understanding the work of the user is the primary target of the research, and that all the misunderstandings should be corrected. This is called the conventional interview phase. The next step of the interview is to clarify the rules. In traditional Contextual Inquiry process it is desirable for the interviewer to interrupt and ask questions and correspondingly for the user to indicate if the time is inconvenient for interruption. This phase is called a transition. The third part of the CI is the actual interview (the contextual interview proper phase), which consist of observation, asking direct questions, suggesting interpretations, writing notes and recording the whole chain of events. Finally the interviewer should wrap up (the wrap-up phase) the interview, ensure that everything is understood correctly and summarize the process. This is the last chance for the user to revise misunderstandings. [4]

Contextual Inquiry is usually conducted by one person and the interviewee. If two people are used, the roles of note-taker and interviewer must be separated. This means that interviewer is leading the discussion and the note-taker does not involve in it. The approximate length of the interview should be two hours. It is also important to get an overview of user's background and demographic information in order to be able to focus on relevant things. It is also important *not to use predefined set of questions*, but to familiarize oneself with the ares of concern in the process. Artifacts offering relevant information about the work, such as cheat sheets or notes, should be also collected, photographed or copied. [27] Reviewing the notes is usually required after the interview in order to ensure their comprehensiveness. It is probable that some ideas or impressions might have forgotten during the interview. [21]

3.2.3 Analysing the data

The data gathered from Contextual Inquiry consist of notes, recordings and possibly artifacts. In order to be able to analyze the data, it has to be in identical format. This is why it might be beneficial to create a summation of the data. It can be done for example by transcribing important notes from the recordings and by describing the artifacts and their use. Once the data is coherent, the analysis can begin.

The analysis of CI consist of three steps. The first step is to set focus for the analysis. It is often the same than for the inquiry itself, but occasionally insights from the inquiry makes the original focus outdated. In this case, the new focus for the analysis need to be identified. The next phase in CI analysis is to choose the data display. There are various of methods available for that, such as affinity and workflow diagrams. After the data display has been chosen the final step is to organize the data in the data display. [21]

The affinity diagram organizes single notes in to higher categories or hierarchies. There are no predefined categories in affinity diagram for the individual notes. Instead, a single note is being used to define a category and then the corresponding notes are being attached to the same category. In other words, the notes creates categories and categories then raises common structures and themes. After the categories or groups are formed, and there is no floating notes left, the groups have to be descriptively named. Then groups of groups are being collected, and thereby hierarchies created. The named groups and hierarchies, and the headings of them, then represents new information in an affinity. [4, 21]

The workflow diagram can be used if the work process need to be tracked and understood thoroughly. At first, the notes from each interview need to be reviewed, after which the flow charts can be conducted. The flow charts should reflect the work process of each individual in every interview. After that all the flowcharts need to be displayed and compared. Finally, a composite work flow diagram (containing the stages perceived in most of the inquiries) can be created. [21]

3.2.4 Remote Contextual Inquiry

The possibility to remotely evaluate the usability of a system was already discussed earlier in this thesis (see section 3.1). Contextual Inquiry can be also implemented remotely with a few modifications on a traditional on-site approach. The remote version of CI is called Remote Contextual Inquiry (RCI). It aims to create a bridge between the users and the developers and is particularly effectively when a project requires feedback from users in distributed locations. RCI can be considered also when the preparation time is limited and the cost is an issue. It can be set up and conducted in less time than the traditional Contextual Inquiry. [10]

In practice, RCI captures the screen of the user working with the software and the usability professional is in contact with the user for example via teleconference and shared screen. Otherwise the activity is basically the same as in traditional CI: User works with the tasks and usability professional observes, records the usage and gathers information about the real-life usage of the system. The analyst also probes and discusses with the user. [10]

The results of the RCI are versatile. Information about the goals of the users and the tasks to achieve those goals are gathered, as well as measurements of the time elapsed and the number of clicks. Moreover RCI provides feedback on layout, content and behavior of the system. [10]

3.3 Cognitive Walkthrough

Cognitive Walkthrough (CW) was developed in the early nineties and it was originally intended to help reviewing 'walk-up-and-use' interfaces, such as Automatic Teller Machine (ATM). CW is a formal usability inspection method for professionals involved in the development process. The key concept behind CW is to use theory as a guide for design review. It is easy to understand and apply and therefore feasible to use in a regular development process. [12, 17] Contextual Inquiry is additional tool for usability engineering and it is easy to evaluate mockups or sketches of designs relatively quickly with it. [22]

3.3.1 Fundamentals of Cognitive Walkthrough

In theory, human-computer interaction process can be described in four steps. Firstly, the user sets the goal for the activity which is to be accomplished with the system. Secondly, the user examines the user interface for available actions. Thirdly, the user selects the action most likely to make progress. Finally, the user carries out the action and assess the feedback from the system. In real-life tasks these steps would probably iterate to achieve subgoals and to complete tasks. Cognitive Walkthrough examines the correct actions to accomplish a task and the four steps to carry out those actions. [22]

Cognitive Walkthrough consist of two phases: preparatory and analysis phases. The preparatory phase contains prerequisites for the walkthrough. First one of them is a brief description of the user and about the knowledge he or she possess. The second prerequisite is a specific description of one or more tasks to be carried out with the system. The last prerequisite is a list of correct actions required to complete the tasks with the UI. The actual analysis will be accomplished in the analysis phase, where every action of every task will be executed and analyzed. In general the Cognitive Walkthrough method can benefit all phases of system's design and development process.[28, 22]

3.3.2 Preparations

Cognitive Walkthrough can be accomplished by using detailed design specification of the user interface which has been received after requirement analysis and functionality definition processes. The walkthrough can be also performed on a paper simulation, minimal prototype or fully functional prototype of the UI. Formally Cognitive Walkthrough evaluates design's ease of learning by exploring it. [28]

The analysis can be done individually or in a group. In a group the process starts with designers presenting the design to a group of peers and it's usually done after a certain milestone in interface design. The designers can then benefit from the feedback and improve the implementation for the next revision. Participants may represent different organizational units and in the evaluation team they have to adopt different roles, such as recorder, facilitator and various kinds of expert roles. [28]

The first step in walkthrough preparations is to describe the users of the system and choose the tasks for analysis. If the background and technical experience of the users are described in the beginning, more details can be possibly revealed in the walkthrough itself. The selection of tasks (for analysis) is critical and should be based on facts, such as requirement analysis, needs analysis or concept testing. The amount of tasks should be moderate and it is important that the set of chosen tasks include some core functionality and some combinations between those core functionalities. Furthermore, to make tasks as concrete as possible, context descriptions must be included. [28]

After the tasks have been chosen, the action sequences for the tasks need to be described. Basically it means that the sequence of actions, which are required to accomplish the task with the UI, are being described. These actions can be as simple as "press the start button" or "write your name in the text field". However, depending on the level of user expertise and user descriptions, actions might also consist of several simple actions which can then be executed as one block. These kind of actions could be for example filling in the register form or going to a specific website. The interface definition should include the prompts preceding all the actions in the task and the interface's reactions to those actions. If the development is already finished, all the information from the interface is available, but if the development process is only in the beginning, paper descriptions are needed. The level of detail in paper descriptions depend once again on the expertise of the users. [28]

3.3.3 Analysis

The Contextual Inquiry analysis phase examines the actions of the tasks and generates a plausible story or a review about the reasons why the users (which have been defined earlier) would have chosen those actions. These stories are based on presumptions about user's expertise and objectives. Sometimes users trust on their problem-solving skills, which is why it is important to understand the problem-solving process in the analysis phase. In order to mimic this process in the analysis phase, four steps should be taken. First, a rough description of the task to be accomplished should be considered. Then the user interface should be explored and actions should be taken according to assumptions users might have. The third step is to observe if the user interface is returning the expected results for each action. The last part of problem-solving process is to assume and define users' next action.[28] In general, the walkthrough or the analysis evaluates if the user is able to select the correct actions with the user interface.

Four criteria can be used in order to assess the ease of performing the correct actions and thereby completing the task. According to the criteria, the goal, the accessibility of the correct user interface object, the match between the label of the object and the object itself, and the feedback provided should be considered while evaluating the system. [22]

3.4 Interaction Sequence Illustration

Interaction Sequence Illustration is a modified usability inspection method. It differs from traditional inspection methods, such as Cognitive Walkthrough and heuristic evaluation, in a significant manner. Unlike other inspection methods, ISI does not function in isolation from system's actual context and users. It conducts the model of interaction from the real-life environment. In case multiple systems are required to perform the user's task, ISI can also focus on many systems instead of just one. [26] In general, ISI combines user-based testing and usability inspection approaches. The method was originally developed for evaluating the usability of Information Technology (IT) tasks carried out in healthcare industry, but there are no defined reasons why it could not be utilized in different environments.

3.4.1 Description of the Interaction Sequence Illustration

Interaction Sequence Illustration focuses on low level analysis of human-computer interaction and exploits the data acquired during the Contextual Inquiry process. However, the inquiry data has to be complemented with documentation about interaction activities, photos and screenshots. There are three objectives for ISI method to handle. The first is to demonstrate how the user perceives the system. The second is to identify and document activities, and to discover problems. The first two objectives forms the basis for the third and more extensive objective, which is to support the user-centered design and development. [26] The strength of ISI lies in the analysis of data and it can be used to compare the interaction sequences of two of or more UI implementations. On the other hand it can provide prominent information on only one UI's interaction sequence.

ISI generates two analysis from the collected data. The first one is an analysis of interaction stages, which divides the whole interaction sequence in to main phases or stages. This analysis is performed based on the inquiries. The second analysis is so called step-by-step illustration. It defines the stages and interaction steps by numbers, photos, descriptions and screenshots. [26] It is basically a well defined and ordered workflow description.

3.4.2 Utilization of the Interaction Sequence Illustration

The utilization of Interaction Sequence Illustration can be simplified in seven steps (see Figure 3.2). First the data need to be collect alongside Contextual Inquiry interview, including inquiry data, possible photos, screenshots and notes. After the data collection the screenshots need to be arranged in the right order and all the superfluous data need to be removed. The third phase is to count the interaction steps based on the screenshots and activity analysis, and to organize interaction steps into stages. Next the screenshots need to be modified and important details highlighted. Finally, the sequence numbers and detailed description texts should be attached to the screenshots in order to give a profound understanding about the actions. The outcome of the method is an extensive illustration of the interaction sequence, information about the usability of the system and about it's effectiveness of use. [26]

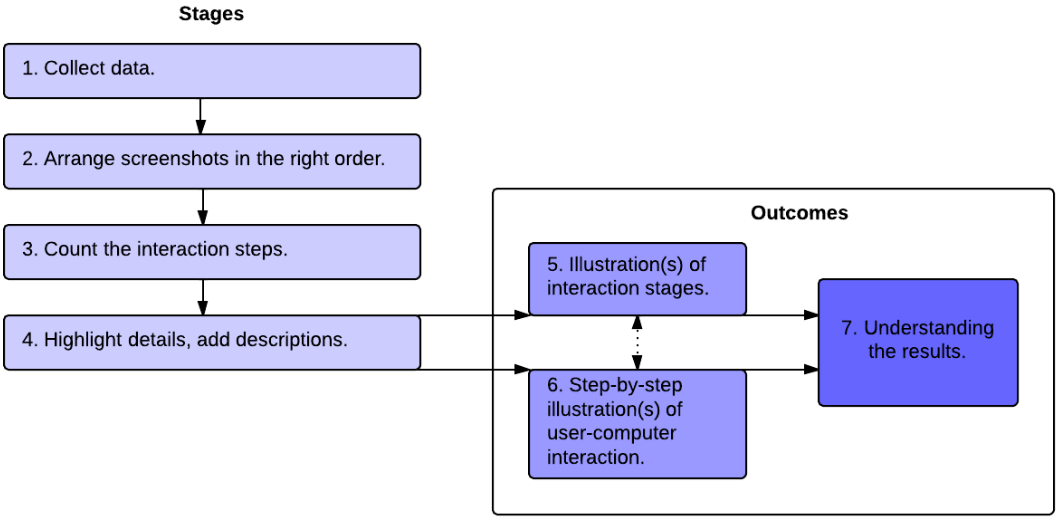

Figure 3.2: Interaction Sequence Illustration stages and outcomes.

3.5 System Usability Scale

In his paper Brooke [6] argued that usability is not any real existing quality, but a good usability artifact is *appropriate to its purpose*. In other words "the usability of any tool or system has to be viewed in terms of the context in which it is used, and its appropriateness to that context"[6]. Still in many cases, context related usability evaluation is not desirable. The reason for this is that a large scale context analysis is usually neither cost-efficient nor practical.[6] In the 1986, System Usability Scale was introduced to respond these challenges by offering "quick and dirty"[6] way to get subjective ratings about the usability of a system. [2, 18] System Usability Scale can be also used as a supplement for another usability evaluation methods.

3.5.1 Description of the System Usability Scale

System Usability Scale (see SUS form) is a ten-item *Likert scale*, meaning that every item consist of the scale of five, ranging from "Strongly disagree" to "Strongly agree". The questionnaire is generally being filled right after the possibility to use the system to be evaluated. The focus should be on immediate responses and too much time shouldn't be given to the respondents. [6] One of the best qualities of SUS is that it's not dependent on any specific technology or user interface type. Thereby it can be used to evaluate the usability of wide range of different kinds of interfaces from traditional desktop user interfaces to mobile web UIs. SUS is also relatively fast to implement by administrators and to use by the participants of the study. [2] `TÄHÄN LISÄÄ!!!rw12 -ARTIKKELISTA`

3.5.2 System Usability Scale in practice

The outcome of SUS is a single value which express the overall usability of the system. The value of the method consist of all the items and none of them are meaningful as such. System Usability Scale can be calculated by summing the score contribution (range from 0 to 4) from each item. Before summing the scale positions, the items 1,3,5,7 and 9 need to be subtracted by one and the items 2,4,6,8 and 10 need to be subtracted from 5. The last step is to multiply the sum of the scores by 2.5 to get the overall SUS value, which will range from minimum of 0 to maximum of 100. [6] The resulting single score is an easy-to-understand measure, and can therefore be discussed with the wide range of stakeholders. [2]

Chapter 4

Implementation of the study

According to ISO standard 9241-210, human-centered design consist a few activities and iterative process (see Figure 4.2). [8] The empirical part of this thesis adapts human-centered design principles and the steps of the process experiment are highly linked to its activities. The rationale for adopting human-centered design is to experiment how standardized human-centered activities fit in with company's software development process. The human-centered design and the selected methods are represented as a part of software development process in figure 4.1.

This chapter contains description about the principles of human-centered design. Applied steps utilized in the research study are described in section 4.2. This chapter will also describe the implementation phases of the experiment.

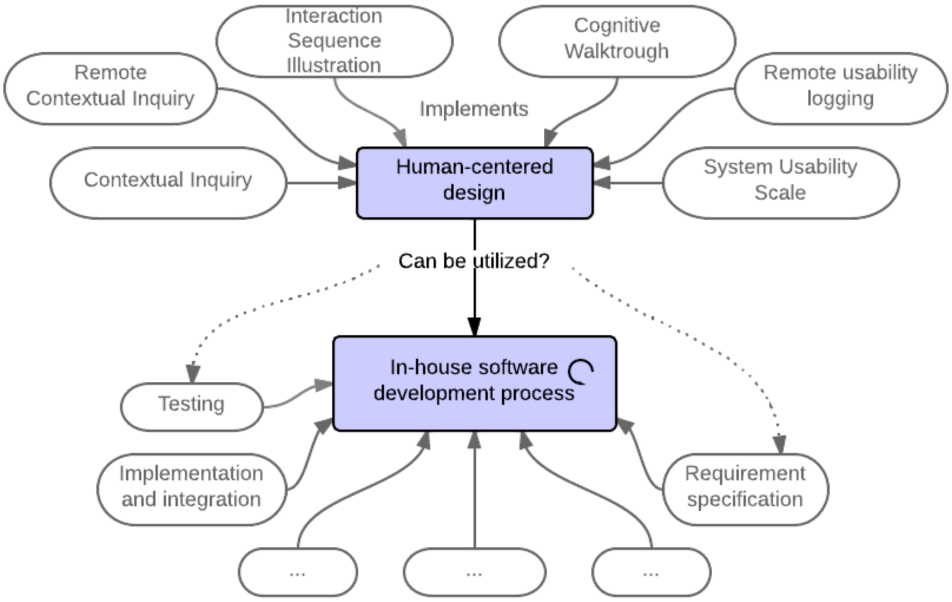

Figure 4.1: Human-centered design as a part of in-house software development process.[8]

4.1 Principles of human-centered design

Human-centered design doesn't require any particular design process, and it functions as a supplement to existing design methodologies. However, there are some principles which should be followed in order the process to be human-centered. Firstly, the understanding of users, tasks and environments should create the basis for the design. This means that all the user groups and stakeholders who might be affected directly or indirectly by the developed system, should be considered. Secondly, users should be actively involved in the design and development processes, in order to provide crucial information about the context of use and the practices of work. The people participating in the design and development processes should possess comprehensive understanding about the work and should be able to represent wide range of users. Thirdly,

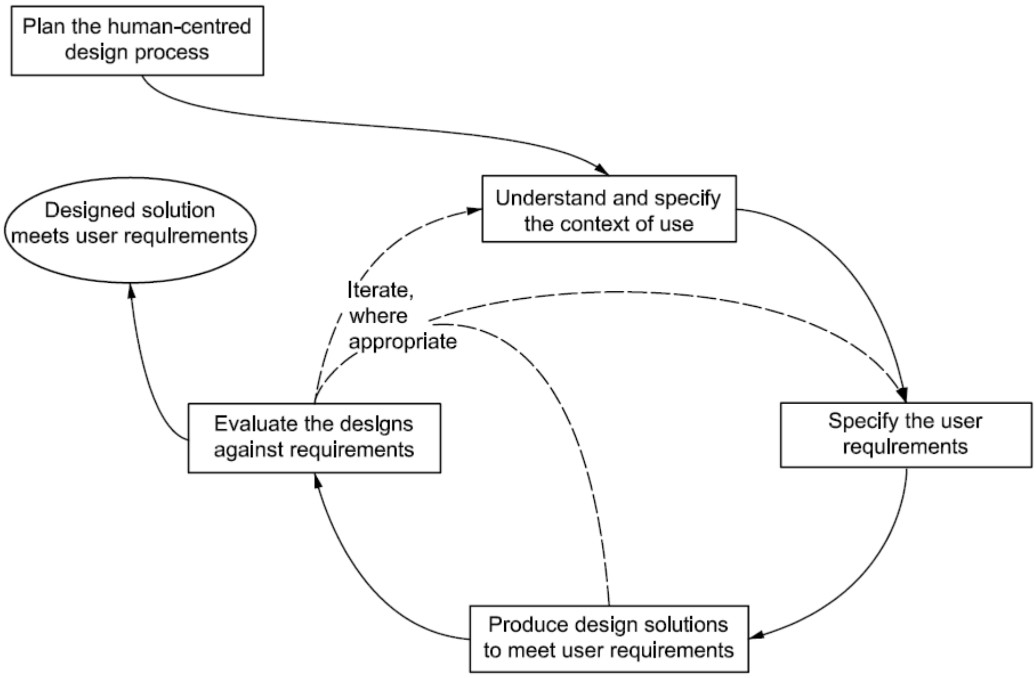

Figure 4.2: Human-centered design activities.[8]

[8]

4.2 Applied human-centered design process

The subscriber company has a local customer service department in every country it operates and they are in charge of customer creation process to the ERP system. The employees of customer services are also maintaining customer information in order to keep them up to date, which requires daily actions to the system, such as editing delivery addresses and invoicing frequencies. Furthermore, the accuracy of customer information in the system is also highly important for the business. Consequently, the customer information processing is examined throughly and used as an example to improve the existing software development process.

This part of the thesis forms the first activity of the human-centered design called *planning the design process*. The section applies the activities of the human-centered design and creates a proposal of methods and procedures to be included in the company's software development process.

The examined ERP process is executed by the personnel highly expertized in the area of customer service and the primary tool for the process is a tailored ERP system. Moreover, the distinctive nature of the business sets additional challenges for the usability evaluation. This is why it is presumable that the process can not be understood thoroughly without understanding the work in practice. Because of the reasons mentioned earlier, Contextual Inquiry is chosen as a method for *understanding and specifying the context of use*. Also, the researcher must have a profound understanding about the context of use, in order to *specify the user requirements*. A qualified system can be implemented only if the user requirements are precisely defined. Many of the methods used in this experiment aims to assist in this crucial state of development. Technically, the process of requirement specification can be initiated while running the Contextual Inquiry. Although, it is essential to remember that the focus in CI should be mostly on understanding the context of use. This thesis uses user evaluation, remote usability evaluation, expert evaluation and quantitative metrics to identify all the user requirements affecting on the user experience and the functionality of the system.

`TÄHÄN SIITÄ ETTÄ CI TOTEUTETTIIN RCI:nä`

Interaction Sequence Illustration method is used in the thesis to identify and represent the interaction steps required to accomplish the customer creation process to the system. The method was chosen because one of the main objectives of this thesis *is to find out how much the efficiency of use can be affected by utilizing the results of usability evaluation*. ISI can be used not only to identify all the interaction steps, but to detect the unnecessary steps as well. If the unnecessary phases can be excluded from the process, it might have significant affects on the efficiency of use and reveal tacit user requirements.

The user experience might have influence on the efficiency of use. This is why the UX of the system should be considered even if the company's distinctive ERP system is developed for the use of experts. The System Usability Scale is being used to understand the level of user satisfaction.

Traditional and continuous usability evaluation would be difficult and expensive for the subscriber company because of its distributed operational environment. Hence, the possibility of accommodating remote usability evaluation to the company's software development process is examined in this thesis. Remote usability evaluation in this context means the remote usability related user interface data logging. The desirable results from using the data logging would be utilizable, location independent usability data, which could help the development of the global ERP system and reveal user requirements.

Expert evaluation is utilized using Cognitive Walkthrough. The objective of CW in the thesis is to identify user requirements which are not distinctive and cannot be perceived with the user evaluations. Cognitive Walkthrough is used to ensure that any crucial requirements will not stay unnoticed.

After the user requirements have been specified, *design solution can be produced according to these requirements*. Analyzed data is applied to create a new user-centered prototype solution. The solution and *the design is then evaluated against the requirements*.

Figure 4.3 illustrates the methods used and the actions taken in the applied design process and reflects them against the actual human-centered design process. Contextual Inquiry, Interaction Sequence Illustration and System Usability Scale are utilized to understand the context of use and the level of sophistication of the current solution. Those methods in addition to Cognitive Walkthrough are applied to specify the user requirements. The SUS and ISI methods measures the existing solution and the data is exploited in analysis and user requirement specification. The design solutions and the prototype is evaluated with Cognitive Walkthrough, ISI, remote usability data logging and System Usability Scale. The final analysis is carried out using the data from the second evaluation.

`COULD BE A PART OF AGILE/ITERATIVE DEVELOPMENT`

Human-centered design stages	Utilized usability methods	Applied design process stages
‣ Plan the human-centered design process.	-	‣ Plan the applied design process.
‣ Understand and specify the context of use.	‣ CI, ISI, SUS	‣ Utilize methods to understand and specify the context of use.
‣ Specify the user requirements.	‣ CW, ISI, RCI, SUS	‣ Use methods to specify the user requirements: 1. Measurement 1. Analysis
‣ Produce design solutions to meet user requirements.	-	‣ Design and produce a prototype to meet the user requirements.
‣ Evaluate the designs against requirements.	‣ CW, ISI, Remote usability logging, SUS	‣ Use methods to refine the design and requirements: 2. measurement, 2. Analysis
Cognitive Walkthrough = CW, Contextual Inquiry = CI, Interaction Sequence Illustration = ISI, Remote Contextual Inquiry = RCI, System Usability Scale = SUS		

Figure 4.3: Relations between human-centered design process stages, methods utilized in the thesis and the implementation stages of the thesis.

4.3 Practical implementation

The practical implementation of the research study was dated between August and NOVEMBER???? 2013. Altogether six users from two different countries participated in remote data logging, Contextual Inquiry and Interaction Sequence Illustration evaluations. They also filled in the System Usability Scale form. Cognitive Walkthrough was implemented by one usability analyst. The users' work experience varied from under a year to a quarter of a century. In every evaluation session the users were situated in their own workstations. `Kuinka monta osallistujaa, millainen ympäristö, missä maassa, minkä ikäisiä osallistujat, kauanko tehneet työtä kuinka testaus sujui, menetelmäkohtaisesti, mitä välineitä käytettiin miten toimi.`
-kaksi vaihetta: ennen ja jälkeen prototyypin
-vaihe1: testi-tietokanta käytössä, jotta pääsi paremmin käsiksi toimintaan, osittain myös tuotanto-kannan kanssa tekemistä, jotta autentisuus saatiin esille (tarkemmat tsekkaukset kirjoituksessa yms...).
-neljä osallistujaa suomesta
-vuodet yrityksessä (noin): 1,21,14,0.5 = avg.9
-suomessa työympäristö

4.3.1 Contextual Inquiry

Contextual Inquiry was executed in three separated phases in the research study. In the first CI implementation the user and the analyst shared the common space, in this case the user's workstation. -ISI ja ci toteutettu kaksi kertaa etänä.

-kesto noin kaksi tuntia
-eteneminen: ensin ci ja lopuksi määritelty interaktio-setti isiä varten. ymmärrys järjestelmästä ja prosessista. käyttäneet paljon, päivittäin.
-analyysi toteutettiin affiniteettidiagrammin avulla.

4.3.2 Interaction Sequence Illustration

-vaihe1 ja 2:toteutettiin interaktio-setin kautta
-pyrittiin mahdollisimman laaja-alaiseen täyttöön esimerkki-tehtävässä.
-analyysi toteutettiin menetelmän ohjeiden mukaisesti.

4.3.3 Cognitive Walkthrough

4.3.4 System Usability Scale

-toteutettiin nettilomakkeella anonyymisti
-osallistumisprosentti.
analyysi vanhaan tutkimukseen peilaten.

4.3.5 Remote usability evaluation

-results -no of clicks

4.3.6 Building the prototype

-technology: .NET, C, WPF, WCF
-using the results from the first round evaluation.
-limitations
-functionality

Chapter 5

Analysis

5.1 Results and comparisons

5.1.1 Overview

-sales department's contribution is required.

5.1.2 Contextual Inquiry

vaihe1: suomessa:57 huomiota, ongelmakohtaa,parannusideaa
-6 kategoriaa

5.1.3 System Usability Scale

-google form
-vaihe1: suomessa tulos: 48.125/100
-vaihe1: suomessa tulos ei välttämättä kerro koko totuutta koska otanta kohtalaisen pieni.

5.1.4 Interaction Sequence Illustration

-interaktioiden ajallinen mittaaminen ei onnistu
-interaktioiden määrä vaihtelee riippuen tilanteesta.aiheutti hankaluuksia, joten otettu keskiarvo.s
-mitkä ovat interaktio stepit ja staget.
-vaihe1: interaktio katsottu videolta ja laskelmat tehty olemassa oleviin asiakkuuksiin syötettyjen tietoihin perustuen. 50 satunnaisotanta antanee ko-

htalaisen selkeän kuvan siitä kuinka paljon prosessi vaatii interaktioita.
-vaihe 1: järjestelmä antaa periaatteessa mahdollisuuden täyttää 83 kenttää per asiakastyyppi + aloitus stepit -¿ käytännössä näin ei kuitenkaan ole.
-vaihe1:sopimusasiakkaalla 50 satunnaisotannan avg. steppien määrä on 23, pienin 9 ja suurin 35 (laskettu tietojen syötöstä tallenna-klikkaukseen.)
-vaihe1:laskutusasiakkalla 50 satunnaisotannan avg. steppien määrä on 14,48, pienin 9 ja suurin 22 (laskettu lisää laskutusasiakas -painikkeen klikkauksesta tallentamiseen)
-vaihe1:toimitusasiakkaalla 50 satunnaisotannan avg. steppien määrä on 12,94, pienin 11 ja suurin 18 (laskettu lisää toimitusasiakas -painikkeen klikkauksesta tallentamiseen)
-vaihe1:uudistusasiakkaalla 50 satunnaisotannan avg. steppien määrä on 15,66 pienin 11 ja suurin 24 (laskettu lisää uudistusosoite -painikkeen klikkauksesta tallentamiseen). myös laskutusasiakas voi olla uudistusasiakas, mikä voi vääristää tulosta hieman.
-vaihe1:yhteensä asiakasinteraktioiden avgt on 66,08 steppiä. tähän tulee kuitenkin lisätä vielä libsysin käynnistykseen liittyvät stepit.

The results of the research may not be suitable for every organization. -Comparison between country offices, between methods -Comparison between individuals (esim. kuinka kauan kesti tietyn toiminnon tekeminen) / Overall comparison (esim. kaikkien koehenkilöiden yhteinen kehitys.) - Interaction sequence illustration (esim. kuinka monta steppiä ennen ja jälkeen)

5.2 Implementation analysis

-Should these methods be implemented as a part of the process or not. -While writing the thesis, contextual inquiry used as part of requirement definition process.

Chapter 6

Conclusions and discussions

jos käytetään hcd:tä ja menetelmiä voidaan päästä eteenpäin umm tasoilla.??

6.1 Future work

-need for more simple methods and more versatile data.

Bibliography

[1] *Usability in practice: how companies develop user-friendly products.* Academic Press Professional, Inc, San Diego, CA, USA, 1994.

[2] Bangor, A., Kortum, P. T., and Miller, J. T. An empirical evaluation of the system usability scale. *International Journal of Human-Computer Interaction 24*, 6 (07/29; 2013/07 2008), 574–594. doi: 10.1080/10447310802205776; M3: doi: 10.1080/10447310802205776; 03.

[3] Bateman, S., Gutwin, C., Osgood, N., and McCalla, G. Interactive usability instrumentation. In *Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems* (2009), ACM, pp. 45–54.

[4] Beyer, H., and Holtzblatt, K. *Contextual design: defining customer-centered systems.* Access Online via Elsevier, 1997.

[5] Botta-Genoulaz, V., and Millet, P.-A. An investigation into the use of erp systems in the service sector. *International Journal of Production Economics 99*, 1-2 (0 2006), 202–221.

[6] Brooke, J. Sus -a quick and dirty usability scale. *Usability evaluation in industry 189* (1996), 194.

[7] Cockton, G. *Usability Evaluation.* The Encyclopedia of Human-Computer Interaction, 2nd Ed. The Interaction Design Foundation, Aarhus, Denmark, 2013.

[8] DIS, I. 9241-210: 2009. ergonomics of human system interaction-part 210: Human-centred design for interactive systems (formerly known as 13407). *International Organization for Standardization (ISO).Switzerland* (2010).

[9] Earthy, J. Usability maturity model: human centredness scale. *INUSE Project deliverable D 5* (1998).

[10] English, J., and Rampoldi-Hnilo, L. Remote contextual inquiry: A technique to improve enterprise software. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (2004), vol. 48, SAGE Publications, pp. 1483–1487.

[11] Fouskas, K., Pateli, A., Spinellis, D., and Virola, H. Applying contextual inquiry for capturing end-users behaviour requirements for mobile exhibition services. In *Proceedings of the 1st International Conference on Mobile Business in Electronic Form* (2002).

[12] Franzke, M. Using the cognitive walkthrough in iterative design. In *Posters and Short Talks of the 1992 SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 1992), CHI '92, ACM, pp. 37–37.

[13] Hartson, H. R., and Castillo, J. C. Remote evaluation for post-deployment usability improvement. In *Proceedings of the working conference on Advanced visual interfaces* (New York, NY, USA, 1998), AVI '98, ACM, pp. 22–29.

[14] Hilbert, D. M., and Redmiles, D. F. Extracting usability information from user interface events. *ACM Computing Surveys (CSUR) 32*, 4 (2000), 384–421.

[15] Jacobs, F. R. Enterprise resource planning (erp) - a brief history. *Journal of Operations Management 25*, 2 (2007), 357–363.

[16] Leon, A. *Enterprise resource planning.* Tata McGraw-Hill Education, 2007.

[17] Lewis, C., Polson, P. G., Wharton, C., and Rieman, J. Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 1990), CHI '90, ACM, pp. 235–242.

[18] Lewis, J., and Sauro, J. The factor structure of the system usability scale. 94–103.

[19] Notess, M., and Blevis, E. Integrating human-centered design methods from different disciplines: Contextual design and principles. In *Proceedings of the design research society futureground 2004 conference* (2004).

[20] O'Regan, G. *Introduction to software process improvement.* Springer, London, 2011.

[21] Raven, M. E., and Flanders, A. Using contextual inquiry to learn about your audiences. *SIGDOC Asterisk J.Comput.Doc. 20*, 1 (feb 1996), 1–13.

[22] Rieman, J., Franzke, M., and Redmiles, D. Usability evaluation with the cognitive walkthrough. In *Conference Companion on Human Factors in Computing Systems* (New York, NY, USA, 1995), CHI '95, ACM, pp. 387–388.

[23] Rogers, Y., Sharp, H., and Preece, J. *Interaction Design: Beyond Human - Computer Interaction.* Wiley, 2011. 2011006430.

[24] Schuler, D., and Namioka, A. *Participatory design: Principles and practices.* Routledge, 1993.

[25] Topi, H., Lucas, W., and Babaian, T. Identifying usability issues with an erp implementation. In *Proceedings of the Seventh International Conference on Enterprise Information Systems* (2005), Citeseer.

[26] Viitanen, J., and Nieminen, M. *Usability Evaluation of Digital Dictation Procedure - An Interaction Analysis Approach*, vol. 7058 of *Information Quality in e-Health.* Springer Berlin Heidelberg, 2011, pp. 133–149.

[27] Wendell, J. B., Holtzblatt, K., and Wood, S. *Interactive Technologies : Rapid Contextual Design : A How-to Guide to Key Techniques for User-Centered Design.* Morgan Kaufmann, Burlington, MA, USA, 200412 2004. ID: 10254657.

[28] Wharton, C., Rieman, J., Lewis, C., and Polson, P. *Usability inspection methods.* John Wiley & Sons, Inc, New York, NY, USA, 1994, ch. The cognitive walkthrough method: a practitioner's guide, pp. 105–140.

Appendix A

SUS form

System Usability Scale

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

Appendix B

Contextual Inquiry: Research plan

Contextual Inquiry practicalities

Focus:

- A customer creation process in the current ERP system is not highly optimized and especially the contact information input actions could be improved significantly.
  1. How the customer creation process can be expedited?
  2. How the number of errors can be minimized?
  3. How editing the customer details can be made easier?

Inquiry Design:

- The subscriber company wants to create a new ERP system. The whole work process (from the need of creating a new customer or editing it emerges, until the Save button has been pressed) needs to be examined in order to enhance the customer creation and contact information input processes.
- The inquiry needs to aim into a more straightforward and partly automated process implementation.

Inquiry in practice:

1. Introducing
    - Ask background and demographic information
    - Make the focus clear for the user
    - Ask permission to record
    - Introduce the method and master-apprentice approach

   The conventional interview. Duration approximately 15 minutes.

2. Stating the rules
    - Set rules
        - User works
        - Interviewer interrupts

   The transition. Duration approximately 30 seconds.

3. Interviewing
    - Observe, interpret, ask questions, suggest interpretations, analyze artifacts and write notes
    - Recall principles of context, partnership, interpretations and focus as guidelines

   The contextual interview proper. Duration approximately 60 minutes.

4. Summarizing
    - Summarize interview and relevant occurrences
    - Ask for corrections
    - Ensure thorough understanding

   The wrap-up. Duration approximately 15 minutes.