Lappeenranta
University of Technology

Anssi Jääskeläinen

INTEGRATING USER EXPERIENCE INTO EARLY PHASES OF SOFTWARE DEVELOPMENT

Thesis for the degree of Doctor of Science (Technology) to be presented with due permission for public examination and criticism in the Auditorium 1382 at Lappeenranta University of Technology, Lappeenranta, Finland on the 16th of December, 2011 at noon.

Acta Universitatis
Lappeenrantaensis 460

Supervisors Professor Jari Porras
Laboratory of Communications Engineering
Lappeenranta University of Technology
Finland

Kari Heikkinen, PhD
Laboratory of Communications Engineering
Lappeenranta University of Technology
Finland

Reviewers Virpi Roto, PhD
Department of Computer Science
University of Helsinki
Finland

Professor Tommi Mikkonen
Department of Software Systems
Tampere University of Technology
Finland

Opponent Janet Read, PhD
Department of Computing
University of Central Lancashire
UK

ISBN 978-952-265-186-0
ISBN 978-952-265-187-7 (PDF)
ISSN 1456-4491

Lappeenranta University of Technology
Digipaino 2011

Preface

The reference applications used in this dissertation were implemented in projects called VIRTAHEPO (2005) and MoMUPE (2006), when the author was a thesis worker and partly a postgraduate student. The dissertation work was conducted during 2008-2011. Writing took place at the beginning of 2011. This work was mainly funded by my work as an assistant and additional funding was contributed by Tekniikan edistämissäätiö in 2009 and 2011. Also Graduate School on Software Systems and Engineering supported this dissertation work by funding one month of work.

My greatest gratitude goes to Kari Heikkinen who has been my mentor from the first moments I start to work for the communications software laboratory. Without your perceptive comments and generous assistance, this dissertation would probably not have been as extensive as it is. I would also like to thank the Head of the Laboratory of Communications Engineering, Jari Porras, whose professional comments at the latter end of writing process were really helpful. Also Päivi Porras provided valuable help especially in statistical methods and analyses. Tokens of gratitude are also given to my co-workers for their generous help,-despite their having areas of expertise outside user experience. Effect of the pre-examiners for the outcome of this work cannot be underestimated, their professional know-how aid me greatly while finishing this work. Lastly, but certainly not least, my loving gratitude goes to my wife, Katja and our beloved daughters, Anni and Nina. You are the lights of my life.

While all the above did what they could do aid me in my research, there were some obstacles. Use of common sense in university bureaucracy is apparently forbidden. In addition mid-way through a project, it is nice to hear that there are only nine months left to finish the dissertation. This is the reason for the dissertation being a monograph rather than a bundle thesis. Initially, the dissertation study was enjoyable, but after the bureaucratic nonsense started to intrude, it became not so enjoyable.

Despite the above-mentioned tribulations, I would make the same decision again since my overall experience was positive. My advice to anyone who is considering conducting a doctoral dissertation is to think carefully before starting. In addition, with the current system of “1+2.5 years and out” researchers should be really enthusiastic about their research topic and have a clear picture about the research and its future progress from the first moments.

Anssi Jääskeläinen

Mikkeli, 11/30/11

Abstract

Anssi Jääskeläinen
INTEGRATING USER EXPERIENCE INTO EARLY PHASES OF SOFTWARE DEVELOPMENT
Lappeenranta, 2011
118 p.

Acta Universitatis Lappeenrantaensis 460
Diss. Lappeenranta University of Technology

ISBN 978-952-265-186-0, ISBN 978-952-265-187-7 (PDF), ISSN 1456-4491

The value and benefits of user experience (UX) are widely recognized in the modern world and UX is seen as an integral part of many fields. This dissertation integrates UX and understanding end users with the early phases of software development.

The concept of UX is still unclear, as witnessed by more than twenty-five definitions and ongoing argument about its different aspects and attributes. This missing consensus forms a problem in creating a link between UX and software development: How to take the UX of end users into account when it is unclear for software developers what UX stands for the end users. Furthermore, currently known methods to estimate, evaluate and analyse UX during software development are biased in favor of the phases where something concrete and tangible already exists. It would be beneficial to further elaborate on UX in the beginning phases of software development.

Theoretical knowledge from the fields of UX and software development is presented and linked with surveyed and analysed UX attribute information from end users and UX professionals. Composing the surveys around the identified 21 UX attributes is described and the results are analysed in conjunction with end user demographics. Finally the utilization of the gained results is explained with a proof of concept utility, the *Wizard of UX,* which demonstrates how UX can be integrated into early phases of software development. The process of designing, prototyping and testing this utility is an integral part of this dissertation.

The analyses show statistically significant dependencies between appreciation towards UX attributes and surveyed end user demographics. In addition, tests conducted by software developers and industrial UX designer both indicate the benefits and necessity of the prototyped *Wizard of UX* utility. According to the conducted tests, this utility meets the requirements set for it: It provides a way for software developers to raise their know-how of UX and a possibility to consider the UX of end users with statistical user profiles during the early phases of software development. This dissertation produces new and relevant information for the UX and software development communities by demonstrating that it is possible to integrate UX as a part of the early phases of software development.

Keywords: **User experience, UX, software development, attributes, utility**
UDC 004.415:004.414.92:159.95

List of Abbreviations

Abbreviation / term	Meaning
ABS	Anti-lock Brake System
ACM	Association for Computing Machinery
ADIRU	Air Data Inertial Reference Unit
ANOVA	Analysis of Variance
CORPUS	Change Oriented analysis of the Relationship between Product and User
CRM	Customer Relationship Management
DRM	Day Reconstruction Method
ENIAC	Electronic Numerical Integrator And Computer
HCD	Human-Centered Design
HCI	Human-Computer Interaction
HFE	Human Factor Ergonomics
I-UxSED	Interplay between User Experience Evaluation and Software Development
IEEE	Institute of Electrical and Electronics Engineers
IM	Instant Messaging
IRC	Internet Relay Chat
ISO	International Organization for Standardization
LINQ	Language Integrated Query
LUT	Lappeenranta University of Technology
MIDP	Mobile Information Device Profile
MMORPG	Massively Multiplayer Online Role-Playing Game
MoMUPE	Mobile Context-Aware Applications and Games
MS	Multiple Sclerosis
MUPE	Multi-User Publishing Environment
NATO	North Atlantic Treaty Organization
OSS	Open Source Software
ROI	Return Of Investments
SIG	Special Interest Group
SMS	Short Message Service
SPSS	Statistical Package for the Social Sciences
SQL	Structured Query Language
UCD	User-Centered Design
UI	User Interface
USAF	United States Air Force
USB	Universal Serial Bus
UX	User eXperience
UXEM	User Experience Evaluation Methods in Product Development

VIRTAHEPO	VIRTuAl Hotspots Enabled by PersonalisatiOn
VUUM	Valid Useful User Experience Measurement
WLAN	Wireless Local Area Network
WPA2	Wi-Fi Protected Access II
XP	Extreme Programming

Table of Contents

1 Introduction

UX (User experience) has many names such as an experience, experience, and overall user experience. Whatever UX is called, the concept has been widely adopted and is used in many fields, e.g. the travel industry, commercials, news, engineering, etc. However, UX is shrouded by a cloud of uncertainty that can be captured by a single question: What is UX and from whose point of view is it seen?

Arhippainen concluded in her dissertation that "*in influences on user experience the key factor is a user*" [9]. This statement may sound obvious to anyone not familiar with UX research, but when it is combined with the following utterances; "*UX is a strange phenomenon*"[55] , "*UX is a very personal and unique phenomenon*"[101] and "*UX is an ambiguous buzzword*"[48] and when demographics, abilities, characteristics, and other aspects of users and user behavior are taken into account, the obvious soon becomes very complicated. All of the above statements are undoubtedly true and illustrate a major problem afflicting UX and UX studies; UX is too vague.

Many definitions for UX can be found in the literature. Twenty seven of these definitions are summarized on a web site *All About UX*[1]. The number of definitions and amount of research conducted in this area, e.g. [96], [135] indicate that clear consensus of the precise definition of UX is missing. One of the many listed UX definitions is from the new ISO (International Organization for Standardization) 9241-210:2010 standard: "*A person's perceptions and responses that result from the use and/or anticipated use of a product, system or service*." [70]. In spite of the establishment of this standard, according to one delegate of the ISO standard working committee, some critically evaluated aspects were omitted [75].

The problem of a lack of a clear definition of UX has been identified in many publications during the last years e.g. [9], [18], [55], and at the time of writing the most recent one [135]. Many workshops like VUUM (Valid Useful User Experience Measurement), UXEM (User Experience Evaluation Methods in Product Development) and I-UxSED (Interplay between User Experience Evaluation and Software Development) have been arranged to tackle or resolve the problem, but most of the solutions and suggestions have been theoretical and concrete methods and proof of concepts are missing. For example, Bevan [18] suggested that evaluating and measuring UX would be simplified if UX and its different perspectives were identified and distinguished and Väänänen-Vainio-Mattila et al. [166] wrote about UX gap between

1 http://www.allaboutux.org/

different communities that try to understand and evaluate UX, and suggested that UX needs to be manageable and measurable.

Until the recent "Demarcating User eXperience" seminar[2] and published UX white paper [135], research in the area of UX has been an ever-growing multidisciplinary field that has expanded in many directions, e.g. hedonic, pragmatic, cognitive, holistic, short-term, long-term, single attribute, etc. Participants of the seminar concluded that UX as a concept is too broad to be useful in practice and suggested that actions should be taken in order to demarcate the field [135]. Still, it needs to be clarified that this suggestion was not signed by all participants of the seminar. This is a significant suggestion and will certainly lead to improvement, but who are the people to take responsibility for this demarcation and how can it be ensured that the drawn lines will be respected. It may be conjectured that even if this 're-focusing' occurs, researchers and practitioners will still use the methods and attributes that best fit their needs.

Within the field of UX, the amount of theoretical knowledge is not a problem, the difficulties lie rather in making this knowledge suitable for real world needs, like software development. Both fields, software development and UX, can be described as huge and multifaceted, so integration is seen as a challenge, but also desirable [79], [134]. The *All About UX* site (footnote 1 page 1) has listed 96 different UX evaluation methods for software development. It should be noted however that the site only summarizes the methods and gives references to the original source. Most of the sources are theoretical and concrete information on ways of utilizing the methods is missing. In addition, the presented methods are biased to the later phases of software development, i.e. after completion of implementation. Yet, the sooner the UX aspect is brought into the software development process the better. Therefore, a long-term wish has been the development of a way to evaluate and analyse UX during the early phases of the software development process [149] [163]. If a method is developed to suit the needs of the real world, it is likely to be commercialized. Attrakdiff[3] is one of these commercialized methods and naturally just the basic version with limited functionality is free of charge.

To the layperson the first thing that comes to mind on hearing the word software is probably some application or game that works on a personal computer, but software exists all around us. For instance dishwashers and cars contain embedded software and even power plant controls are software based. In other words, there is software somewhere behind most devices. All software has some purpose, various direct or indirect users, and variable use contexts. All these aspects have an effect on the UX requirements that are set for the software. If UX requirements are not met, the consequences can be frustration, anger or similar negative feelings, which may in turn lead to further actions such as; browsing the Internet for a solution, reading the manual, or even abandoning the software or device. In the most severe cases the consequences can be devastating as in an example presented by Cooper [26]:

> Flight 965 Boeing 757 aircraft was approaching Cali and the pilot action was needed for selecting the next radio-navigation fix, called ROZO. After entering R

2 http://www.dagstuhl.de/en/program/calendar/semhp/?semnr=10373
3 http://www.attrakdiff.de/en/Home/

> to a navigation computer, the computer listed nearby navigation fixes starting with R. Pilot picked the first one since its latitude and longitude appeared to be correct. By following the directions given by the flight computer, pilot crashed the plane into a granite peak at 3000 meters. Navigation fix that the pilot picked was ROMEO instead of ROZO [26].

The example given above, despite being caused by human error, was in fact an indirect problem in the flight computer software. It can be argued that the software worked flawlessly, the problem was actually in usability, the user interface, functionality, expectations, etc. The software was not directly responsible for the accident but indirectly it was the cause since it only gave facts and did not inform the pilot that the chosen radio-navigation fix was a fatal one. Even a simple hint that the given navigation fix was unusual or odd, might have saved the passengers and crew members.

1.1 Authors' path to UX research

During his master's thesis (2005) the author participated in the VIRTAHEPO[4] (VIRTuAl Hotspots Enabled by PersonalisatiOn) project, which was a joint project involving the Technology Business Research Center of Lappeenranta University of Technology, the Youth Department of the City of Helsinki, Nokia Research Center, Tunturi Oy, and the Finnish MS (Multiple Sclerosis) Society. This project awakened his interest in UX and user aspects of software engineering, though Lappeenranta University of Technology did not possess previous expertise in this area.

The main goal of the VIRTAHEPO project was to blur the line between virtual reality and the real world [164]. The author was assigned to a MS Society case, in which the main focus was described as the development of an ”*Online service for MS-patients to strengthen social support from experts, but also from a peer group and increase motivation to do physical exercises as a part of therapy.*” Based on this aim, a multipurpose meeting area called the *Activator* for Finnish MS patients, their therapists and nursing personnel was designed and implemented.

MS is a central nervous system disease and symptoms of MS vary from difficulties of movement, tactile and visual disturbances, speech and swallowing difficulties, to exhaustion following even minor effort [46]. Physical exercise is really important in self-treatment so every possible action is usually taken by the treatment personnel to encourage patients' to take care of their own well-being. A healthy way of life and maintenance of good physical condition with suitable forms of physical activity are the most effective ways of self-treatment. The importance of physical exercise was the main motivation for the design and implementation of the *Activator*, in co-operation with occupational therapists and physiotherapists [81]. The main idea of the *Activator* was to offer MS patients enjoyable moments together by providing different tasks and brain teasers with suitable amounts of physical movement in combination with real information from the surrounding areas. *Activator* was developed in co-operation with occupational therapists so the abilities and wishes of the target users were known. Unfortunately, the physical ability of MS-patients to use computers or pointing devices of MS-patients was non-existent, and therefore this group could not be considered the

4 http://virtual.vtt.fi/virtual/virtahepo/index.html

primary target users. Occupational therapists were chosen as responsible for controlling the *Activator* while MS-patients just sit by, watch and give suggestions. During development, the main focus of the occupational therapists was on the content of *Activator* and not how things should be presented, how accessibility should be considered, etc. Therefore, the actual design for UX and implementation were based on authors considerations and judgments. This probably affected the UX of both the occupational therapists and MS patients. Four screen captures of the *Activator*, Lappeenranta section, are presented in Figure 1. The VIRTAHEPO project gave valuable information about end users, their wishes and needs, and how they should be considered during a software development project.

Figure 1: The *Activator*

On completion of master's thesis, the author worked as a software developer in a project building mobile multi-user games for young people using the Nokia MUPE (Multi-User Publishing Environment) platform [83]. The work was part of the MoMUPE (Mobile Context-Aware Applications and Games) project, which was a joint venture between Nokia, Lappeenranta University of Technology, Helsinki Institute for Information Technology, Tampere University of Technology and the VTT Technical Research

Centre of Finland, aimed at developing new and innovative application ideas and implementing fully functional prototypes on the MUPE platform.

During the MoMUPE project a number of games were developed, but the one called *Greenhouse* is the most significant one for this dissertation and this game will be used as a reference application. In *Greenhouse*, which is presented in Figure 2, the idea is to build a greenhouse. Players need to buy land, pick materials and select heating, lighting, irrigation, etc. and start growing and selling goods. Naturally the choices made during the greenhouse building have an effect on the growing conditions which need to be suitable for the propagated goods. The value of the goods in the market is based on supply and demand as well as the reputation of the farmer (player). It is possible for the players to play their game without having any contact or confrontation with other players, but interaction was made possible by offering a way to launch a sabotage attack against other players' crops or greenhouses, as well as the possibility to check the current top five growers. The demographics of the target users were known, but lack of knowledge and know-how in UX led to designing for UX based on authors own interests and assumptions. The MoMUPE project gave valuable information about another group of end users and about the importance of end user presence in development.

Figure 2: MUPE game *Greenhouse* on Nokia N95

Actual contact with the target users in the MS Society case came very late and the main intention of this meeting was to demonstrate a functional version of the *Activator*. The outcome of this meeting was suggestions for a great number of time-consuming modifications such as the possibility to enlarge things on the screen or use text-to-speech. Later on in the project, further corrections and new features be implemented were suggested for an application that was nearly completed. While some corrections and modifications were made, due to deadlines, all new features could not be included, it being evident that the users would clearly have benefited. In the MUPE case, contact with end users' never came, due to the final part of the whole project and MUPE being transferred to Nokia Beta Labs[5]. Therefore, feedback was only based on comments from personnel involved in the project and the actual UX of end users of author's MUPE games remains a mystery.

The author was responsible for the concepts of own applications and games designed and implemented during projects. With hindsight and current experience in UX, it was realized that it would have been extremely beneficial in both cases, *Activator* and *Greenhouse*, to have had access to some sort of knowledge about the target group's abilities, interests and views of UX. Some feedback was received from project colleagues, but the actual needs and wishes of the target groups were never directly

5 http://betalabs.nokia.com/apps/mupe

heard. These experiences formed the idea about considering UX and its attributes as early as possible during the software development.

1.2 Research objectives, claim and contribution

Path described in Section 1.1 as well as related research described in Chapters 2 and 3 can be transformed into the research question (Q):

- Q: *How to enhance UX considerations in the early phases of software development without big investments?*

The aim of this dissertation is to present UX knowledge from UX professionals and end users with different demographics and abilities as it pertains to software development. Authors' interest, research question and the missing consensus about UX provide the rationale for the following three main objectives (O) of the dissertation:

- O1: *Raising the UX knowledge among software developers.*
- O2: *Considering the attributes of UX by utilizing statistical user profiles in the early phases of software development.*
- O3: *To prototype and test a utility that assists in meeting O1 and O2.*

The claim formed from the above research question and objectives is:

- **The Wizard of UX utility increases software developers' know-how, understanding and ability to consider UX by providing UX information and target users' view on the importance of UX attributes.**

This claim will be proved by combining knowledge of software development projects (mainly VIRTAHEPO and MoMUPE projects) with research information from the fields of UX and software development. This practical and theoretical know-how is supplemented with UX information from UX surveys conducted as part of the research.

This research is cross-scientific by contributing to two different communities; the UX community and software development community. This dissertation provides:

- *A link between end users, UX professionals, software development, and UX theory.*
- *A designed, prototyped and tested UX utility for early phases of software development, called the Wizard of UX.*

1.3 Research methods

Primary research according to [109] can be divided into three types, observations, interviews and surveys. This research utilized quantitative surveys to collect original data and no evidence was found that similar data exist elsewhere, which according to Nunan [127] means it can be considered as primary research. Collected original data can

be seen as one part of the starting points of the constructive research [66], which is the second utilized method of this research. The practically relevant problem underlying this constructive research is the need to be able to consider different attributes of UX in the early phases of software development projects. Theoretical understanding for this problem stems from the fields of UX and software development combined with the effect of user demographics and abilities. In the light of the problem and prevailing theory, UX surveys are constructed and the results gained are used to prototype the *Wizard of UX* utility, which acts as a solution to the problem. The solution is tested with real software developers, and its applicability is examined with a UX industrial designer.

1.4 Scope

Despite the recent efforts at demarcation [135], UX is currently a growing multidisciplinary field, and it would be complex to study it as a single entity. The main scope of this dissertation is limited to the attributes of UX and raising the awareness of software developers in understanding what kind of UX the target user group appreciates. Children, elderly people and people with disabilities are not included as end user groups of this dissertation, but these groups will be considered in future work. Modern software can be considered as a complex entity which might contain many problems. Not all problems are directly related to end user UX and are therefore outside the scope of this dissertation. Also software development in bigger development companies is ruled out from the scope of the research since they have their own established ways to consider UX. Therefore, the main object of this research is individual software developers and small development companies that do not possess the knowledge or resources to consider UX. Nearly 100 methods for evaluating and analysing UX during software development exist, but minority of the methods are applicable to the early phases of software development. The main scope of this work focuses on UX gap described by Väänänen-Vainio-Mattila et al. [166] in the early phases of software development and aims to give people responsible for considering UX a way to take end users' perspective into account and to raise general awareness about UX. Finally, too many definitions for UX already exist and this dissertation will not offer a further one.

1.5 Structure

The structure of this dissertation is presented in Figure 3. The main outcome of this dissertation is in the center of the figure with surrounding elements describing the main features of chapters.

Chapter 2 gives a literature study of UX and presents the historical background and current trends in UX research. It discusses the adaptation of UX to a vast number of fields which has led it to take the form of an umbrella term which now requires demarcating to be actually useful. Chapter 2 concludes with discussions and questions arising from issues considered in the chapter.

Chapter 3 discusses UX in relation to software development. The discussion is related to different development processes and the possible effects of different people included

in the development. In addition, the possible consequences of bad UX are demonstrated. To conclude this chapter, different alternatives to evaluate and analyse UX during the various phases of software development are presented, as well as their suitability for evaluating and analysing UX in the early phase of software development.

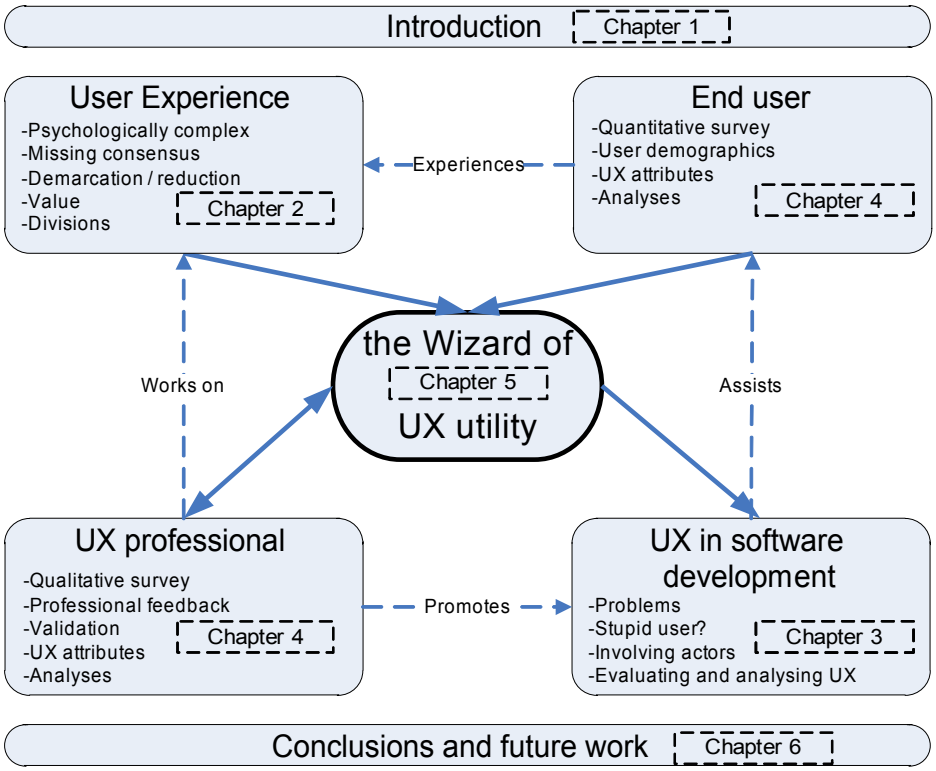


Figure 3: Structure of the dissertation

Chapter 4 presents the designed and conducted surveys used to gather UX knowledge from UX professionals and end users. The chapter presents the chosen respondents and provides justifications for the surveyed demographics and abilities. Analyses of the results are presented and the results are compared based on gender, age and ability to use computers. The chapter concludes by combining the results for all surveyed demographics and abilities.

Chapter 5 presents the design, current structure and functionality of the *Wizard of UX* utility which was designed and prototyped as a proof of concept application for this dissertation. Two reference applications, one software development oriented course and an industrial UX designer were used for testing the *Wizard of UX*.

Chapter 6 presents conclusions, and proposes future work that needs to be conducted. The dissertation concludes with, some final words about authors hope for UX.

1.6 Authors' original publications

Although this dissertation is the form of a monograph, reference applications, the idea for UX utility and preliminary results from the surveys conducted have been presented in earlier publications. Short summaries of the referenced publications [81], [82], [83] and [84] are summarized below and the author's contribution detailed.

- [81] Jääskeläinen, A. Heikkinen, K. Designing, Implementing and Testing Experiental Multi-User Virtual Community With MS-Patients. International Journal on WWW/Internet 5, (2007), 147-164

    - The motivation of this paper was to extend the meaning of independent physical and mental practicing as a part of occupational therapy and thus promote the quality of life of MS-patients. The author's contribution to this paper was in designing and implementing a virtual community called *Activator*, which was constructed based on user and organisational requirements and tested by two different pilot groups formed from MS-patients. Kari Heikkinen gave valuable insight into UX research and experiences, which extended the original conference publication. Some of the first UX attributes are identified in this publication and are used to evaluate the *Activator*.

- [83] Jääskeläinen, A. Lautamäki, J. Analysing Context-Aware Service Development under MUPE Platform. In Proceedings of Eight International Workshop on Applications and Services in Wireless Networks, ASWN '08 (Kassel, Germany, 2008) pp. 26-34

    - This paper describes context-aware service development under the MUPE platform. The author's main contribution to this paper was in designing and implementing the *Greenhouse* reference application as well as writing about half of the paper. Although *Greenhouse* was designed and implemented, it was decided not to include it in the publication since it only used one server side context (temperature).

- [84] Jääskeläinen, A. User eXperience: Tool for developers. In Proceedings of Interact 2009 Doctoral Consortium, (Uppsala, Sweden, 2009), 888-891.

    - This publication presents the idea of a UX in early phases of software development which is the problem addressed further in this dissertation. In addition, the idea of a UX tool to fill the target area is introduced. Work behind this publication and the feedback received from the doctoral consortium form the basis for the UX attributes, *Wizard of UX* and UX database discussed in this dissertation.

- [82] Jääskeläinen, A. Heikkinen, K. Divergence of User eXperience: Professionals vs. End Users. In Proceedings of International Workshop on the Interplay between User Experience and Software Development, I-UxSED 2010, (Reykjavik, Iceland, 2010).

- Preliminary results from the conducted surveys were presented to the community. The author was solely responsible for the paper with Kari Heikkinen working as a mentor and proof reader. The paper discusses the variance of understanding of different aspects of UX among users with divergent demographics and abilities. In addition, results from end users are compared with results from UX professionals.

1.7 Terminology

- End user. In the scope of this dissertation, end user refers to both university students and upper secondary students that were used as sampling of end users of software.

- Software process. Software project should follow the selected development process.

- Software project. The project where software is built.

- UX attribute. In science, the term attribute refers to a characteristic of an object or entity. Therefore UX attribute is seen as one characteristic of user experience.

- UX professional. A person who possess academic publications from the field of UX or is well known by the UX community.

2 User Experience

This chapter gives an overview of UX, distinguishes it from experiences and presents the current research status, definitions, viewpoints and demarcations of UX. Definition, in theory, is easy since the ISO 9241-210:2010 standard defines UX as follows: "*A person's perceptions and responses that result from the use and/or anticipated use of a product, system or service.*" [70]. However, according to Jokela [75], who was part of the ISO 9241-210:2010 standard working committee, the definition may not be the best possible. Apparently the definition and meaning are not that easy since UX is an umbrella term [67]. For some UX might equal with UI (User Interface), for some it might be usability, and for some it might be anything vaguely related to users and technology. The following are a selection of UX views from various sources:

- UX is dependent on the subject, object, and interaction between these two [101].
- No amount of professionalism in UX can substitute for our being personally involved [4].
- UX varies between situations and may change over time [58].

All the above statements strongly emphasize the role of the user and his or her demographics and abilities as well as the current context of the user. Hassenzahl [58] states that the psychological complexity of UX cannot be underestimated. The author agrees with this statement and promotes it by presenting a real world example.

> My beloved daughter who is at the time of this photo a year and a half owns a Moomin moped, which is clearly designed to be a sit 'n' ride toy. I have seen four different children driving with this toy and only the oldest one (four years old) sat on top of it and used her legs to kick the moped forward. I tried to demonstrate this same behavior to my daughter, but she actually pulled me away from the moped and showed me the correct way to drive it. (see Figure 4).

Figure 4: 'Correct' way to drive Moomin moped

All three children from 'just learned to walk' to three year old have used this toy in exactly the same way. After

observing their behavior, it becomes clear why they used the 'correct' way to drive the moped. The physical reason is that their short legs and lack of coordination do not yet allow them comfortably to sit on top of the moped. Another reason might be a psychological one; the four year old driver did not seem to enjoy the moped as much as the younger ones, since it is impossible to build up speed while sitting on top of the moped and obviously driving without speed is unstimulating.

What can be said about the UX of driving the Moomin moped? Was it poor for the younger drivers because they did not use it as it was designed to be driven? Most certainly just the opposite if something can be observed from the facial expressions of the Moomin moped users. On the other hand, the oldest driver used it as it was apparently designed, but based on authors observations the UX for her was not good at all and she quickly abandoned the moped. This example is in line with the statement by Hassenzahl "*There is no guarantee that users will actually perceive and appreciate the product the way designers wanted it to be perceived and appreciated*" [58].

Was the example given above about experience or user experience? Where is the line between these two? In the scope of this work, it is considered as UX since the user was able to use or manipulate the moped. Some might disagree, and this is one of the problems with UX research; clarification is needed of what is UX and what can be considered to belong to the more extensive area of experiences. Pine & Gilmore [129] wrote that experiences are multidimensional and dependent on the level of participation and the type connection with the experience. They did not present any separation between experiences and UX but claim that active participation in an environment in which the user is immersed is an experience. This sort of environment might be a virtual reality headset, paintball game, etc. [129]. In this dissertation such an experience will be called as user experience, since the user is in control of at least some of the elements in the experience. Thus, if someone is a passenger on a roller coaster it is only an experience, but if person is the operator of the roller coaster and driving it for the first time, then it is user experience. Roto [139] has taken a similar approach in her dissertation when she separates UX from experiences: "*I claim that user experience is a special case of experience, where the person can use a system, with or without a purpose. Using means that the user not only senses the system, but also has the opportunity to manipulate or control the system. The system is a product, object, or a set of them; service systems often involve a human being such as a librarian. If there is no system at all, or if the person cannot control the system, we should use the term experience instead of user experience."* [139]. In this dissertation, the term UX is used in accordance with the basic definition given by Roto.

2.1 From experiences to UX

User experience as a field of research is young, but plain experiences have been designed long before the Common Era, e.g. ancient Greek comedy and different forms of theatre are the first forms of designed experiences. Stone Age cave drawings may be considered experiences, despite their actual meaning being anything from borderlines to 'I was here' to a sanctum marker. Millenia later, many great writers and poets, like William Shakespeare designed experiences for people in the form of love, sexual passion, tragedy, comedy, death, etc. Still, the attitudes towards experiences have not

always been flattering. For example the Scottish moral philosopher Adam Smith, the founder of modern capitalism and economics, stated in 1776: "*The labour of a menial servant, on the contrary, adds to the value of nothing*" [148]. By menial servant he means unproductive labourers and according to him some of the most frivolous professions were: "*churchmen, lawyers, physicians, men of letters of all kinds; players, buffoons, musicians, opera-singers, opera-dancers*". Work conducted by these professions according to Smith is "*unproductive of any value, and does not fix or realize itself in any permanent subject*."[148].

2.1.1 First thoughts

The basis for modern UX study was provided by the American psychologist and philosopher, John Dewey, whose pragmatic thoughts about experiences in art, education and nature caused experience theory to enter the mainstream around 1920-1930 [33], [34]. For Dewey the essential conditions of life, such as breathing, eating and warmth, experienced through a person's senses were the determinants of experience. Aesthetic experience, according to Luojus [101], who follows the ideas of Dewey [33] is a common name for specific qualities that are highly valued and an ideal form of experience towards which everyday experiences are striving.

2.1.2 Influence of war

During the First World War the development of sophisticated devices made great advances, especially in the area of warfare [106]. This naturally led to increased interest in human characteristics and abilities, as humans were needed to operate these new devices. Typically the research was conducted with a form of trial and error which continued until an appropriate candidate for a pilot, etc. was found [106]. Rumor tells that Russians selected their tank operators with the following criterion: Anyone who is small enough to fit the cramped quarters of the tank became an operator. Greater consideration of human factors began in the latter stages of World War Two when it was noticed that human interaction with sophisticated technical devices is not an easy endeavor. For example, effective placement of control knobs and more accessible displays in cockpits were studied, which can easily be equated with modern day user interface research. After the World War Two the USAF (United States Air Force) published a 19 volume summary of their research into HFE (Human Factor Ergonomics) conducted during the war [106]. During the Cold War, many research laboratories on both sides conducted human performance and engineering research, leading to the development of branches like human-engineering, psycho-physics and aviation psychology. Clearly, this research was not made public since it was related to military technology. At the same time, civilian industry also established HCI (Human-Computer Interaction) research groups in areas like reliability and logistics, which eventually led to the integration of human factors into system design [106]. As the Cold War thawed, the military saw an opportunity to co-operate with academia and their research gradually became more public.

In 1954 Abraham Maslow wrote The Third Force [51] dealing with the basic needs of humans and psychological aspects of human behavior. He began this work during the early days of World War Two and the target was no more or less than; "*I wanted to*

prove that human beings are capable of something grander than war and prejudice and hatred."[51]. At that time there were several competing psychologies, but Maslow wanted to integrate these various truths into a whole truth. He also felt that existing theories could not solve human problems and could not explain all verified human behaviors. Instead of accompanying or extending existing theories, he was sure that by combining subjective and objective parts a great deal more about human nature could be rationalized. An important finding of Maslow [51] was that if the subjective approach is ignored, some human behavior remains meaningless.

In Maslow's analyses, basic human needs were divided into three different categories; physiological needs, basic needs and growth needs. Human beings are motivated by the first two, which are species-wide, apparently unchanging, and genetic or instinctual in origin. The most powerful needs are naturally the needs for physical survival, which influence human behavior, but only as long as they are unfulfilled. Physiological needs are food, liquid, shelter, sex, sleep and oxygen. Maslow [51] states that when those are satisfied, higher needs like safety and security emerge, and when those are met, growth needs assume importance. He listed truth, goodness, beauty, aliveness, individuality, perfection, necessity, completion, justice, order, simplicity, richness, playfulness, effortlessness, self-sufficiency, meaningfulness, self-esteem, esteem by others, and love & belongingness as growth needs which are all equally important [51]. Despite the fact that this list is more than 50 years old, lots of similarities to modern HCI and UX study can be seen, suggesting that human aspects of UX attributes are integral to UX research.

2.1.3 UX in the media

Many companies have noticed the value of UX as can easily be seen from the news, advertisements and from the Internet. The following news quotes were collected from the online archives of the New York Times[6], BBC news[7] and Helsingin Sanomat[8].

- *"We focused on money and Facebook focused on growing the user base and **user experience**."* (Mr. Dewolfe, Myspace)
- *"Apple's ability to create products that do not fill an obvious need, but through attention to design and **user experience**, produces something that delights the users and challenges conventions."* (Co-founders of Ideacodes)
- *"We've created a great **user experience** with the smartphone and we think this really separates Orb TV from the others in the space like Roku and Google TV".* (Mr. Costello, Orb TV)
- *"Our intention is to develop a $150 smartphone that is similar to iPhone **user experience**." (*Chinese telecommunications equipment giant Huawei)
- *"Nothing kills the successful adoption of new technology better than a poor **user experience**."* (Damian Saunders, Citrix)

6 http://www.nytimes.com/
7 http://www.bbc.co.uk/news/
8 http://www.hs.fi/english/

- *"Its all-important **user experience** looks to have brought some genuinely new thinking to a smartphone market in which all operating systems feel somewhat similar to use"* (Tony Cripps, Ovum's)

- *"Smart location bar of Firefox is the biggest **user experience** change... since tabbed browsing."* (Mike Schroepfer, Mozilla)

- *"The holy trinity in the mobile phone business refers to the combination of hardware, software and services. Nokia has all three and by combining them the company plans to open up a new kind of **user experience**.* (Ben Woods, CCS)

All the above items use the term "user experience" but what is actually meant by the term? It can be assumed that, in the Firefox case it is mainly about usability, functionality and usefulness, and has nothing to do with, e.g. aesthetics or privacy. In the Huawei case, the usage probably means a larger set of attribute that has something to do with UX. On the other hand if the term would be used for the *Greenhouse* case described earlier, it would mean graphics and ability to play with mobile phone. The above quotes illustrate the multifaceted nature of UX, and how it is used but not explained, forcing the reader or hearer to create meaning to the term.

2.1.4 Identifying the economic value

In the modern world, the value of experiences is clearly visible from news and advertisements as the above section presents. Experiences are considered great sales promoters in many companies, which try to improve sales of their products by adding something extra to the package, e.g. collect five codes and get a free ticket to a movie. One of the most important areas of business where experiences are used is the travel industry, e.g. in the UK a travel agency Canterbury Travel[9] promotes trips to Lapland with the following phrase: *"Lapland holidays and particularly a Lapland Christmas holiday is a magical experience for adults and children alike."*. Safaris Iceland[10] sells their wilderness safaris with as *"a once in a lifetime experience"*. It is more like the norm than the exception that a travel agency uses the word experience. The Lapland Centre of Expertise for the Experience Industry has even produced a handbook for experience tourism agents [159]. The document handles issues like combining fact and fiction, producing meaningful experiences, considering cultural aspects, etc. They also sell their own series of Articles on Experiences. Sward and MacArthur [154] suggest building a whole business strategy around UX, which requires the UCD (User-Centered Design) community to broaden its perspectives. They claim that this embedding might give sustainable competitive advantage, but will present challenges like integrating UX with business disciplines and requiring the UCD approach to be adopted by the whole organization [154].

The economic importance of experiences has been acknowledged by the well-known experience experts, Pine and Gilmore, in their book [129]. They illustrate the power of experiences by using a simple coffee-bean example, starting from the commodity that is sold by a harvester who gets 0,50€/kg. When the same coffee is sold in a nice package

9 http://laplandexperience.com/
10 http://www.safaris.is/

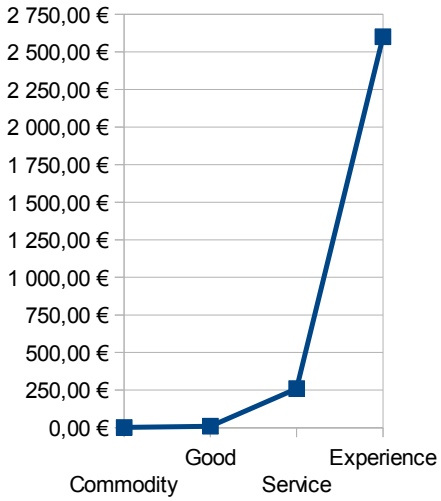


Figure 5: Price of a coffee / kg

in a grocery store as a good, the price is 4-10€/kg depending on the brand. When a gas station owner buys that package and brews a coffee, it is a service and will cost 0,5-2€/cup. In Finland, a basic coffee package weighs 500g and is enough for approximately 65 cups of coffee. Using this as a basis for calculation, a gas station owner receives 65-260€/kg. Finally, if the same cup of coffee is sold, for example in a cafeteria near the St Mark's Square in Venice, the cup might cost up to 20€ (2600€/kg). Naturally there are other expenses than just the price of a coffee package, like salaries of employees, rent of the place, etc. Nevertheless customers are more than willing to spend this money because of the location, atmosphere and experience of the moment with someone they love. Figure 5 present this increase in value.

The above example is about a simple coffee bean, but the value of user experience has also been identified in software development [163], [166]. In software development it is possible to follow e.g. user-centered development process to take users into account, and the software development process might include designers from many areas, like UX, UI and interaction. Unfortunately, not all companies or developers possess such resources. While companies might be aware that UX should be considered, they just do not possess enough time, money, personnel or knowledge to achieve this goal. However, the success or otherwise of software is always dependent on the end users and no amount of expertise in design can guarantee success. Software development should try to see UX from the end user point of view, but the problem is to transfer that view to an understanding of how other parties in the development process see things.

Although the economic value of UX is undisputed, the questions remain how much is it worth and how much is it worth investing in UX. This issue can be transformed into a ROI (Return of Investment) of UX. When referring any commercial activity, profitability is always a valid question. If the primary target group is large enough and their UX according to evaluations is fine, is it economically wise to even take account of the UX of secondary or tertiary target groups? Naturally, there is no financial ROI with freeware applications and games like *Greenhouse*, but ROI still exists in some other form like fame and reputation. Furthermore, even in these non-financial cases, developers need to consider issues like whether it is worth making the application in a way enabling it to be used even by visually impaired users. In the case of the *Greenhouse* application, it was decided not to consider visually impaired users, since by default those users would be only a tiny minority.

2.2 UX - Still a mystery?

UX has evolved greatly since Adam Smith's times and has branched out such that currently the concept covers a very wide area. As mentioned earlier, although ISO has produced a standard and definition of UX, even those who participated in the standard preparation working committee admit that it leaves a lot of open questions [75]. Moreover, academic UX professionals do not seem to support this definition and cannot reach a mutual agreement about the UX issue [96] [135]. The UX research site *All About UX* (footnote 1 page 1) has listed 27 different definitions from various sources and, not surprisingly, their list is not complete. Everyone seems to have their own opinion about what UX is and what attributes or aspects should be included under the umbrella term.

Multiple books and many publications concerning experiences and UX also draw attention to this issue, but little consensus exists [139], [96], [59]. There is not even consensus whether UX can be both negative and positive. Nevertheless, the concept, be it termed user experience, experience, overall user experience, an experience or something else, has been widely adapted to almost every imaginable context and it is a widely used term in the Internet. A Google search with "*superior user experience*" leads to 1 480 000 hits (07/05/11) and if the word superior is dropped out, hit count jumps to 69,8 million. According to Buxton [22], this is a problematic situation since when a word means almost anything or everything; it actually means nothing. Extending the idea further, a sentence from software metrics comes to mind; you cannot control what you cannot measure and you cannot measure what you cannot define [45].

At the Dagstuhl seminar (2010), some UX professionals admitted that the point had been reached where UX is too broad a concept to be actually useful in practice. They suggest that instead of trying to identify every aspect of UX, concentration on demarcating the field of UX is needed. This was not a common agreement since some researchers do not want to limit the research field [30]. Even before this proposal for demarcation, attempts to classify and handle UX with different approaches had been made. UX has been sub-divided into 'an experience', 'experience', and 'co-experience' [47]. Forlizzi and Battarbee explain 'Experience' as: "*the constant stream of "self-talk" that happens while we are conscious. Experience is how we constantly assess our goals relative to the people, products, and environments that surround us at any given time.*" [47] 'An experience' can be articulated or named, "*has a clear beginning and end and often inspires emotional and behavioral changes in the experiencer"* [47]. Co-experience includes the social context and occurs "*when experiences are shared with others or created together*" [47]. Roto, on the other hand, sees overall UX as an extension to UX; "*the overall user experience is formed out of use case experiences and perceptions and information received outside the use cases. The overall user experience affects the user experience of the next use case*" [139], which follows the idea by Mäkelä & Fulton Suri [116].

It is not only academic research that works around UX, e.g. Nokia have defined their own UX elements, by mapping compatible UX attributes into them. These elements are utility, usability, social value and enjoyment [137]. Utility and usability are directly taken from the commonly accepted pragmatic side of UX and social value was picked

based on brand slogan "*Connecting People*". The remaining unmapped attributes Nokia placed under the enjoyment element. Their intention was to define a set of elements that would be applicable for all Nokia products and would reflect the brand core, but they admit that it might be beneficial if each product had at least its own target level for each element [137].

2.3 Clarifying the mystery

During the past decade number of potential UX models has been proposed to clarify the cloud of uncertainty around UX [55], [93]. At the subsections below UX models are presented with following divisions, hedonic – pragmatic, temporary – long-term, and other methods.

2.3.1 Hedonic vs. pragmatic

A common way to categorize UX is to use a division into hedonic and pragmatic sides. In some cases the hedonic side is called emotional, holistic or experiential, and pragmatic side termed, functional or instrumental, but the meaning behind the differing terms is nearly the same. The hedonic side according to Hassenzahl [58] emphasizes individuals' psychological well-being and supports the achievement of be-goals. These be-goals are commonly recognized as stimulation, identification and evocation [97], [58]. Stimulation means that the product must be stimulating to use, for example, it has to offer users new impressions, insights and opportunities because all individuals have a 'need' for personal development. Individuals want to be seen by others in a special way and therefore they need to be able to identify themselves through physical objects like web pages. Thus a product must have a way to communicate this identity to others. Evocation basically means reviving memories of past events, like an old computer game. The pragmatic side is concerned with the product's utility and usability in relation to the task at hand, as well as its perceived ability to support the do-goals [57], [137]. So, the pragmatic product will offer an effective and efficient way to complete the task. An example will clarify the differences. Someone might own a chainsaw that does its job well, is cheap, has a three year warranty and is just fine for their needs. In this case, the chainsaw is pragmatic. If someone has a high end DeWALT chainsaw from winning a contest in which a neighbor lost, then the chainsaw is hedonic because it has emotional value.

Hassenzahl [58] uses a four locker categorizing system to describe product characteristics emerging from hedonic and pragmatic attributes. An extension of his system with concrete examples from author's own life is shown in Figure 6.

> The combination of weak hedonic and weak pragmatic characteristics leads to an unwanted result, which in this case would be an Opel Vectra station wagon. It was spacious, powerful (3,2l V6), had xenon lights and nice to drive on the highway, but during the cold winter the breather tube of the crankcase ventilation system froze twice and it nearly caused an oil fire in the engine compartment. In addition, the suspension was dreadfully noisy while driving on bumpy roads. A combination of strong hedonic and weak pragmatic characteristics would lead in this case to a Peugeot 206 SW, which worked as our wedding vehicle. It also was our first car bought with a loan, was nice to drive and powerful enough. After our first daughter

was born, there was not enough room for all of us plus our three cats, so its pragmatic value was not strong anymore. In the lower right corner a Toyota Avensis can be found, which is our current car. It is great to drive and has enough space for all of us, but we do not have any emotional attachment to it, nor does it make the neighbors jealous. Therefore, to us, it is not hedonic, only pragmatic. The final locker in the upper right is the 'dream box,' which contains an Audi RS6, one of the world's most powerful (and expensive) family cars. It would be pragmatic, with space, the latest safety devices and probably many other desirable features. Certainly this car would also make almost every neighbor jealous so therefore it would also be hedonic.

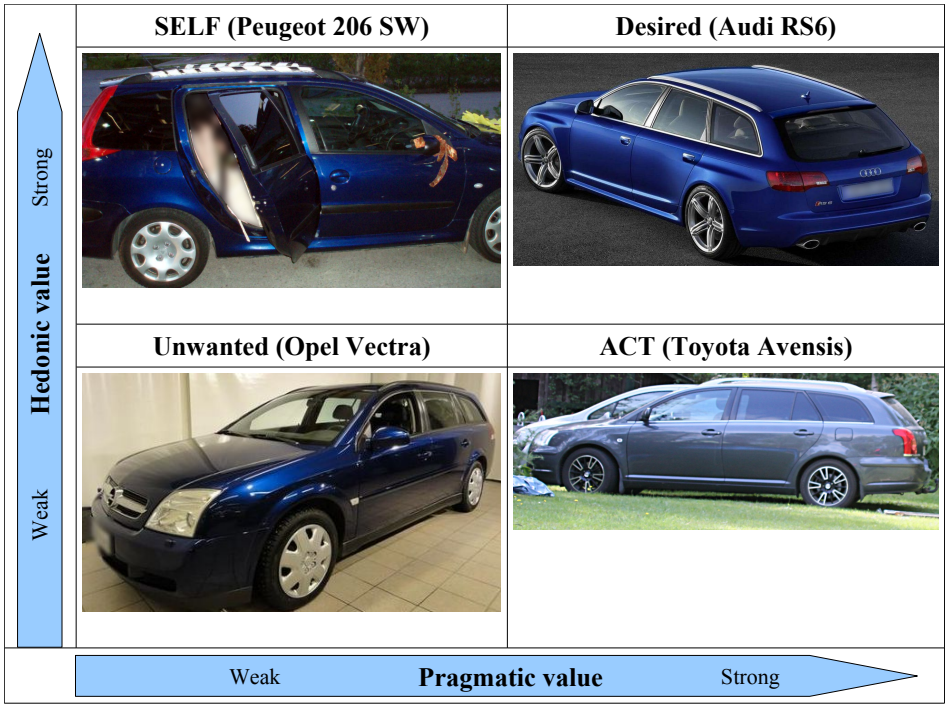


Figure 6: Hedonic and pragmatic examples

Novak and Schmidt studied the importance of hedonic stimulation while collaborating with large displays in a travel agency. They found that in general positive user attitude toward the system was strongly related to hedonic attributes in both the client and the travel agent [126]. Arguments about the Hawthorne effect (things that are measured, evaluated, etc. will be more effective since the work moral of test people is increased) [76] and the effect of novelty of their large touch sensitive display can be put into question. It would be interesting to see whether their test results can be replicated once the system has been used for years. Nevertheless, their finding about the importance of hedonic stimulation undoubtedly has validity. Väänänen-Vainio-Mattila and Wäljäs [167] have developed both pragmatic and hedonic evaluation heuristics for web service user experience. They conclude that while of interest, the task is challenging since many

attributes belong to both the hedonic and pragmatic sides. They also concluded that not all heuristics are usable in every situation and that some situations require additional heuristics.

Interpretation of pragmatic and hedonic sides and their dependency or in-dependency vary, but an important question is raised; is it possible to design emotions or are emotions too ephemeral [58]? The problem with the division into pragmatic or hedonic attributes is the fact that it is sometimes difficult to decide if something is pragmatic or hedonic. In addition, many attributes affect both sides either directly or indirectly [167].

2.3.2 Temporary vs. long-term

The majority of UX research concentrates on temporary UX, which happens at the moment when the user experiences the effects of his or her actions and reflects on this experience. According to Law et al. [135] this is the core area of UX. Still, anticipated UX [135] is formed before the actual usage and have effect to UX. In this dissertation, anticipated UX is considered with attributes brand and expectations.

Three different UX factors; context, user and system [55], affect both temporary and long-term UX. Temporary UX is closely tied to the context aspect, which is the broadest of the three aspects. This aspect contains the social context, task context, temporal context and physical context that come from the outside system and user [138]. This complexity makes it the most difficult one to fully take into account. Consequently, the next day the temporary UX of the same system exactly and the same task might be different due to e.g. different temporal and social context.

The user aspect is strongly dynamic because users are dynamic. This aspect considers current mental state of user e.g. emotions and motivation, as well as current physical state of user like accessibility [138]. For example, if a user has just performed vigorous gymnastic exercises and his or her hands are shaky, a good UX is unlikely to be achieved with a system that requires a lot of accuracy with a mouse or other pointing device.

The system aspect is everything that is designed and implemented in the system that naturally influences the UX of the user. Further on, according Roto [138] system aspect includes also all systems that affect to system under investigation. Attributes like accessibility, aesthetics, functionality, interaction and stability are categorized under this aspect. It also contains objects that users have added to the system, which might, for example, be a background picture of a loved one.

Long-term UX is formed cumulatively and can be viewed as the stable background that the temporary UX slowly molds over time. Long-term UX is sometimes called as overall UX [139] or cumulative UX [135]. Regardless of the precise term used, the area has been recognized, and experts seem to think that UX research over a longer time span might affect findings since the temporal and dynamic nature of UX cannot be seen in a few minutes or even few hours [135]. In theory, temporary UX always has some effect on long-term UX, but little research in this area has been reported, possibly due to the fact that study of long-term UX takes a lot of time and academics rarely have

enough resources or time to conduct such longitudinal studies. Still few retrospective methods like DRM (Day Reconstruction Method) [88] , iScale [87], CORPUS (Change Oriented analysis of the Relationship between Product and USer) [170] and UX Curve [93] have been developed. On the other hand, if a company were to conduct such a study they could gain considerable benefits for future product development and most certainly would not be willing to share their knowledge with competitors.

Luojus states in her dissertation that temporal UX alone might not be enough as a basis of design [101]. Moreover, she also finds that bad temporal UX did not affect overall good UX of users of a fitness heart rate monitors. In fact, she states that the effect of short-term UX on long-term UX is almost zero [101]. This raises the question: How much bad short-term UX is needed that it affects long-term UX? This is a similar question to the problem with sand grains and a pile; if you add one grain of sand to table, is it a pile? When you add another one is it a pile already? Where is the line between individual grains and a pile? The relationship between the grains and the pile is similar to that of short-term bad UX and long-term UX; everyone is an expert about themselves.

2.3.3 Other models and views

Følstad [49] advocates simple measures and adhoc models rather than complex models and a mass of attributes. He supports his statement with examples like Amazon book ratings and YouTube video ratings, which both use just one measure to indicate the quality of a target [49]. This is an effective way to rate something that already exists and could be used, for example, as an evaluator of long-term UX. However, the approach does not provide answers to the question of how to know what to fix if the ratings are low. Følstad's method [49] is suitable for summative evaluation that is used for selecting the targets that require more observation e.g. formative evaluation. By utilizing this approach, the amount of work could be reduced radically, since there is no need to run thorough tests on every target.

Holistic viewpoint have also be used to consider UX. This view is interested in human needs, view of life and consciousness. In other words, the human is seen as an entity built from physical, mental and social elements. The holistic approach can be traced back to Dewey [33] and [34]. His holistic view links together a wide range of areas, such as; emotions, memory, plans, environment, thought processes of the user, consciousness, interests and reactions to current conditions. Dror [37] for example has taken the holistic view and states that not only the technology and its performance that should be considered in technical development. Instead, wider factors of the context must be taken into account during design and development to ensure the success of new technology. He claims that if these critical factors are not taken into consideration, the result might be a failure of the new technology even if it is technologically superior compared to others. Luojus [101] in her dissertation presents a similar approach. She claims that the dissertation by Roto [139] uses unnecessary and unhelpful approaches that will lead to cognitive reduction of experience due to the lack of a holistic view. Nevertheless, on the same page, she admits that some methods might be practical for product development, but not suitable for seeking scientific knowledge [101].

Finally UX can also be considered from the cognitive viewpoint since cognition is based on brain structure and how it functions [150]. However, going into the cognitive processes of human mind is far out of the scope of this dissertation.

2.4 Discussion

Currently, UX is a subject of debate, particularly the proliferating number of quasi-synonymous terms, aspects, attributes. For example, at a NordiCHI workshop in 2010[11] it was stated that accessibility is not part of UX, an opinion with which the author disagrees since it does not account for users with disabilities. While accessibility may not be an issue to users without disabilities, it will most certainly be a UX issue to users with disabilities or illnesses that make normal usage difficult or even impossible [81]. In the discussion, no common agreement was reached among the academics present. According to Law et al. [96] when industrial UX professionals are taken into the debate, they naturally have opinions that suit their particular area. Not surprisingly, these opinions are dissimilar to those of academics.

A common approach in academic literature is to reduce UX to a number of factors, processes or attributes [153] which can be measured utilizing further sub-divisions. Whenever such reduction is done, an important question should be considered, namely; where is the line between reduction and loss of important information? In spite of this risk of losing important information, this dissertation reduced UX into 21 attributes originally presented at [82]. Currently, the attributes are presented as individuals without any connection to each other.

The scope and definition of UX is not only an issue in the academic environment, but also of relevance to industry. For example, Nokia defined it own UX elements, while nevertheless stating that it would be good if every product would have their own target levels for these elements [137]. The question arises whether there can ever be truly universal UX elements, in view of the fact that, in addition to being pragmatic, UX is also a strongly holistic concept that covers all aspects of experiencing a phenomenon.

This chapter has raised many interesting questions: How can a word mean something to an average software developer when even UX professionals are unable to decide? This is a major problem and major issue in current software development. How to design, implement or evaluate “user experience” when its meaning varies? Or even if UX is specified from the developers view, is it known how clients or end users see it, especially if there is no possibility to meet them? This problem can be mirrored to research question presented in section 1.2). In this context, Lord Kelvin’s dictum may be cited: "*When you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind*". On the other hand, it needs to be verified that the results from measurements are useful, valid and meaningful since: *"The route from theory or law to measurement can almost never be traveled backwards. Numbers gathered without some knowledge of the regularity to be expected almost never speak for themselves. Almost certainly they remain just numbers."* [92].

11 http://users.dsic.upv.es/workshops/i-uxsed10/background.html

In conclusion, UX is traceable back to the user. Naturally, aspects of the current context, used device or system, etc. influence UX, but the understanding and realization of these aspects is dependent on the demographics, abilities, characteristics, etc. of the user. For example environment may be the same for two users standing side by side, but the users might still experience it with completely differently.

3 UX in software development

This chapter considers UX from the software development point of view and presents the current situation of UX in different design methods and software development models. It explains how the different actors in a software development project affect the UX experienced by the end user and what can be the consequences of bad UX. Finally, ways to evaluate and analyse UX during the different phases of a software development project are presented and their suitability for the early phases is analysed.

The basic idea of software development is to build a solution that satisfies some need. This need may come from many different sources, such as from the programmer, from the boss or from the client. When the basic idea is transferred to the larger scale merely meeting needs is not enough; projects need to be completed with low costs, in a limited development time, and with a high-quality [72], without forgetting user experience.

3.1 Path from independent to integrated

In the early days of software development client requirements were collected and analysed, but concepts such as HCI or UX did not exist. The first actual programmable computer, ENIAC (Electronic Numerical Integrator And Computer) was invented 1948[12], but software development as an area of science is merely 50 years old. Academic interest in this field started in 1968 when the term software engineering was used in an international conference sponsored by the NATO (North Atlantic Treaty Organization) Science Committee. Conference topics included management and methodologies, design, system evaluation, and documentation [119].

In 1990, when the IEEE (Institute of Electrical and Electronics Engineers) 610.12-1990 standard [65] about software engineering was released, the concepts of HCI and UX existed, but the situation was relatively unchanged. The standard defined software development as: "*The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is the application of engineering to software*." [65]. Despite the standard, Sutcliffe and Wang [152] stated that integration between HCI research and practice with methods in software development is necessary, but currently nonexistent, and Shaw stated that; "*Software engineering is not yet a true engineering discipline, but it has the potential to become one.*" [147]. The problem with the IEEE definition of software engineering is the same

12 http://www.seas.upenn.edu/about-seas/eniac/

as in the UX case; the definition is too wide and imprecise to be useful without correctives for a particular case.

Considering UX during software development has become more important, since many modern technical devices are loaded with software [53]. Cars, washing machines, remote control devices, and even to door locking systems, e.g. iloq Privus[13] contains software which is hidden from the users, but still responsible for the functionality.

Journals about software development for example IEEE Software[14] and IEEE Transactions on Software Engineering[15] exist and UX publications can be found from the proceedings of conferences such as CHI[16] and nordiCHI[17]. These two fields have been independent areas of research, but attempts have been taken in order to integrate HCI and software development. For example Hedberg and Iivari [60] proposed a model which added a human level, which should contain a HCI core team, usability designers, usability evaluators and non-technical users. Their intention was to highlight the role of HCI specialists in OSS (Open Source Software) development projects [60], since it is common that in OSS projects, usability and UX are neglected and software is produced by engineers for engineers [8], [121]. Da Silva et al. [27] have studied the combination of agile methods and UCD (User-Centered Design); they conclude that there clearly is a need for further studies considering combination of these two fields. Najafi and Toyoshiba [117] combined user experience design with agile development to improve product usability. They conclude that both are iterative so they naturally complement each other, but such an approach will require full co-operation and collaboration between different teams and team members.

Apparently considering HCI during software development is nothing new. However, it needs to be realized that design methods and HCI are only parts in creating the overall UX and ways for evaluating and analysing UX are biased in favor of the latter phases of software development. Furthermore, according to Väänänen-Vainio-Mattila et al. [166], UX evaluations are often handled by and within the design group and not as part of the overall development process. This naturally decreases the trustworthiness of the results since the designers are evaluating their own work. To aid software development to take users into account, design methods, such as HCD (Human-Centered Design) and interaction design, have been created.

3.2 Human-Centered Design

The 'father' of UCD is Donald Norman, who describes the term in his book 'The Psychology of Everyday Things' [124] in the end of the 1980s. The term is currently included in the ISO 9241-210 standard [70] as HCD. The standard defines four activities which are essential in every project that follows HCD; requirements gathering, requirements specification, design and evaluation. Still the standard only offers general guidelines for design and it approaches development from a high level of abstraction

13 http://www.iloq.fi/privus/
14 http://www.computer.org/portal/web/software/home
15 http://www.computer.org/portal/web/tse
16 http://www.chi2011.org/
17 http://www.nordichi2012.org/

[73]. The main idea of HCD is to put the users at the center of the whole process, by communicating directly with the user. In theory HCD takes users into account, but does not describe any actual methods and does not specify at which point or and how users should be considered. Therefore, HCD can be considered a development philosophy rather than an actual development model.

Jokela [73] approached this issue by proposing a KESSU 2.2 model that extends the standard into seven activities with an outcome driven way to describe each activity with five subsections; purpose, characterization, outcomes, requirements and methodological guidance. Many third parties have worked on this standard and it is combined with actual methods and different areas of research and industry, e.g. focus groups are suited to requirements gathering and usability testing, and card sorting and participatory design are suited to design activity [169]. Bullinger et al. [21] have combined a HCD method with traditional participatory design in architectural design in order to enable planners to involve end users in immersive and spatial prototypes. Thimbleby [160] has studied HCD and its suitability in cases where the importance of user special needs must be taken into account.

3.3 Interaction design

Alben [5] succinctly expresses the essential question of interaction design in one sentence: "*How does effective interaction design provide people with a successful and satisfying experience?*". She defines the ACM (Association for Computing Machinery) design awards criteria for quality of experience as the outcome of the following areas; understanding of users, learnable and usable, needed, mutable, effective design process, appropriate, aesthetic experience and manageable [5]. While the list is comprehensive, it actually provides few concrete answers. For example, needed or learnable by whom, and with what demographics? How are the understanding of users and mutability to be measured? Furthermore, the paper only considers design, it does not comment whether the design can be followed throughout the development.

Despite being a fairly new field, interactions have been designed for centuries, though not with this particular name. The intention of interaction design is to create meaningful relationship between users and products by creating a solution to a known problem. In other words, interaction design involves trying to understand and alter the way people do things, feel things and how they think about things [89].

Interaction design includes many research techniques where the key is to investigate users and their environments to make it possible to design for UX. If being successful, interaction design greatly influences the end user on an emotional and personal level [89]. After research, all aspects of development need to be combined in a way that is efficient and suitable for the company and for the end user or client. A common way to make all acting parties to realize the requirements is to use, for example personas or other forms of user profile that are built based on the target group(s) [89]. However, personas are created based on thorough investigation of users and their environments and smaller companies and individual developers do not possess the resources to conduct the required actions.

In optimal situation, interaction designer(s) participate in the implementation and system testing in order to be able to verify implementation or on-the-fly modifications that might affect overall UX. When utilizing rapid development method such as Scrum this in theory is possible, but does it happen in reality. In addition it needs to be kept in mind that interactions are only a part of the multifaceted concept of UX.

3.4 Software development models

Every development model brings its own benefits and challenges to the software development, but it is beyond the scope of this work to delve deeply into the different models. Common models, which roughly contain the phases presented in figure 7, are briefly introduced to clarify their relevance to UX and software development. According to [110] the models are divided into sequential models which consist of models similar to the waterfall model, and flexible models, which consist of agile models and other non-sequential models.

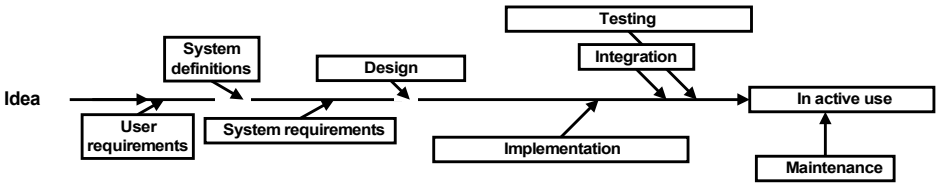


Figure 7: General software life cycle

3.4.1 Sequential models

One of the best known models, and in fact the only true sequential approach, is the waterfall model, which was introduced by W. W. Royce in 1970, although not with the above-mentioned name. Royce [140] presented this model as a defective, non-functional model. In the unmodified waterfall model, every phase must be completed before the next phase can begin. This feature has received most of the criticisms. Critics state that, e.g., a client can change requirements during the development process, which leads to redesign. Also, designers cannot predict or know all the problems that will occur during the implementation phase, which again will lead to a redesign. Despite the criticisms waterfall became the US military standard for military grade software projects [36] and was soon adopted by NATO countries as well. This is probably one reason it is still widely used, despite the fact that its inventor stated that as a process it simply does not work [158].

Efforts have been made to enhance the basic waterfall model. For example, Iqbal and Rizwan [68] suggest that applying the Pareto Principle, also known as the 80/20 rule, to the waterfall model will make it more efficient. They state that 20 % effort can be focused on critical activities, which according to their findings will lead to 80 % productivity and performance. Lott [100] suggests that the waterfall model is a usable method, especially from the management point of view. The author states that it is a lot easier to estimate risks, time and costs when development happens in clearly defined stages, in contrast with agile methods. "*With $25,000 our best team will work with you*

for two weeks to find opportunities, plan your future tasks and build a plain prototype. At the end of the period we will (or will not) offer you a proposal for doing the next stage, for price X" [100].

None of the original steps in the waterfall model relate directly to UX and very few scientific papers have combined the waterfall model and UX. Sun [151] propose the waterfall model for experience management and knowledge management, they claim that their method provides a new approach to data, experience, and organization management. In [78] and [79] Joshi et al. have integrated HCI activities into software development processes and measured the effectiveness of the integration. In their paper they write that: "*the modeling phase of the waterfall model should include detailed user interface prototyping, formative usability evaluation of the user interface, and refinement of the prototype*". However, their final conclusion is that the index of integration is greater in projects that use agile models than in projects that use the waterfall model. It should be noted that usage of a sequential model does not preclude use of the UX knowledge and UX utility provided by this dissertation e.g. before the requirements phase.

3.4.2 Flexible models

The scrum model [157] and XP (Extreme programming) [15] are non-sequential development models which are used in modern software development. The scrum model was first described by Takeuchi and Nonaka in 1986 [157]. Scrum is based on sprints (cycles) that last 2-4 weeks and the intention of every sprint is to build a more complete product. In the beginning of a sprint, a set of features are chosen from the prioritized set of high level requirements that are included in that particular sprint. Every sprint is performed by a scrum team, which is typically made up of 5-9 people and remains unchanged during the sprint. XP was invented 10 years later by Kent Beck [15] and like Scrum it is based on frequent releases in short development cycles. The main activities in XP are coding, testing, listening and designing.

Scrum and XP have been adapted in many different areas, but the literature does not specify user experience roles like UI designers, content writers or user researchers for either software development life cycle model. Therefore, in theory, it is unclear how to integrate UX methods and practices into agile development environments [2], [20]. For example in Scrum there are three only different roles; team, scrum master and product owner. Scrum team is responsible for everything that is done during each sprint, which includes UX related tasks. Kollman et al. [90] conclude that all agile methodologies are fast, iterative and incremental, so therefore UX and UCD people need to adapt their working methods to the agile world. Good understanding of agile methodology and active participation in the process are needed to facilitate a UX practitioner's ability to work in an agile context [90]. Sy [155] came to similar conclusion, but stated that the flexibility of agile methods is seen as a big advantage, but also as a big challenge.

Real-world examples of combining the agile model with UX also point in the same direction. For example, Paypal[18] decided to develop a large project using Scrum, while the rest of their development stayed with the traditional waterfall model. They found

18 https://www.paypal.com

that UX persons faced many challenges and pitfalls during the project before their working habits had evolved to meet agile needs [20]. A similar example comes from the Salesforce[19] enterprise CRM (Customer Relationship Management) marketplace. In 2006 they transitioned many product teams from a waterfall to a scrum method. After the transition UX persons were dissatisfied. Common complaints were: Being assigned to too many teams; too much time being wasted on meetings; not enough time being available to do the actual work; and a lack of focus on the big picture. At the same time, R&D persons were satisfied, so the UX persons just had to adapt and evolve. After one year of adapting and evolving a satisfaction rate of over 80 % was achieved [44], but unfortunately it is untold how many UX persons leave the company during the process. The solutions in both cases were similar replicated those found by researchers. In the Paypal case it was found that the UX teams should work ahead of the development teams as a separate scrum team and that design vision sprints should be added to the normal sprint cycles. After adoption of the enhanced working methods, the authors state that using scrum brought many benefits [20]. In the Salesforce case, co-operation, working ahead, and utilizing prototypes were seen as suitable solutions [44].

Apparently agile methods are more suitable for designing for UX during the software development than sequential models. Both of the presented agile methods bring the end user closer to the development, but many challenges remain in integrating agile and UX communities so that they can work effectively together. In addition, both of these agile models are development models, meaning that these can be used after some starting points for the project are decided, while the

3.5 Problematic software?

There exist design methods and development models that assist software developers, but still the Internet is full of articles about software failures, malfunctions, and problems. For example, the change from the year 2009 to 2010 caused Windows Mobile 6.1 and 6.5 systems to send SMS (Short Message Service) from the year 2016 [54]. In another case, Skype Windows clients suffered a critical failure that lasted about 24 hours. The failure itself was caused by overloaded servers, which caused certain Windows clients to crash [130]. A further example is from 2010 when some owners of the Toyota Prius experienced inconsistent braking when the ABS (Anti-lock Brake System) was activated. Toyota resolved this issue by updating the ABS control software [161]. In the reference application *Greenhouse*, the server side starts to crash spontaneously after being online for more than 10 days without use. None of the examples above lead to injuries or loss of life but such examples also exists.

> In 2008 a Qantas Airbus A330 started a sudden nosedive while traveling at 37,000 feet. ADIRU (Air Data Inertial Reference Unit) generated very high, random incorrect values for the flight control computer, which caused the autopilot to disconnect. Despite the autopilot being off, the flight control computer still commanded key controls and according to ADIRU data started pitching down at a maximum of 8.5 degrees. Pilots quickly regained control and no one was killed in the incident, but more than 50 passengers and crew members were injured [1].

19 http://www.salesforce.com/

> During the Gulf war, a Patriot missile was not launched against an Iraqi Scud missile due to a software error in the system's clock. The radar system spotted the Scud, but because of the error in the system it looked at the wrong part of the sky and did not find the Scud again. Therefore, the system assumed it was a false alarm and removed the detection from the system. The fact was that the system had been continuously used longer than it was supposed to be used, but still 28 soldiers died and about a hundred were injured. The error was fixed day after the incident. [120].

Almost all software products are released with known defects [85]; why is this? Promised release date which equals no time to fix the problem(s) undoubtedly have effect. Therefore, at release to bypass the problem, it is only stated that the problem is known and will be fixed later. Naturally, these defects will not be made visible in commercials nor demonstrations, although the software might contain a small *readme* file which contains the list of known defects. This file often cannot be accessed until the package is opened and opening the package means that the client is no longer allowed to return the product. When the multifaceted concept of UX is combined with problematic software, it is easy to state that eventually there will be effects on UX. Naturally not all software problems cause UX problems. For example a following hypothetical situation:

> A defect causes an infection which crashes some background system. This is noticed by a monitoring system which immediately restarts the background system and sends a notification to system administrator. This happens in fraction of a second and is totally transparent to the end user of software who does not even know that the background system was crashed.

Still, if software problems are not transparent to end user, those will affect on UX. As a result, user might even change the software vendor or operating system. In the workplace, on the other hand, employees have no other options but to use the software chosen by the company or find another job. Employees are, in other words, paid to tolerate bad software which may make the pain a bit more bearable. As an employee they might feel frustrated and unhappy, but not complain, yet their UX with the software is not good.

In a book 'Bad Software' [85], Kaner and Pels try to explain why the situation is so bad and how consumers should assert themselves. The authors claim that software companies rarely provide real warranties for their products and that their technical support is rarely anything, but inadequate. Jokela [74] has partially considered this flaw in his publication about usability requirements in calls for tenders. He states that by defining usability requirements in calls for tenders, the company which responds to the tender commits to deliver what they promised. If too many unsuccessful attempts happen during the end user tests, company is responsible for the corrections, without extra costs to the client [74].

3.6 Or stupid users?

If the software is not the problem is it the users? Programmers, in particularly, easily seem to judge users as stupid when they struggle with their solution during observed usability tests [50]. Even worse, if a programmer is in the lead, (s)he might refuse to make changes to a solution because (s)he thinks that users are too lazy to learn how to

use his/her "superior" design [26]. Founder of Microsoft, Bill Gates has said: "*Let's face it, the average computer user has the brain of a Spider Monkey*"[20]. This quote is widely used in the Internet. This same "feeling" is indicated by the results of a survey study; according to more than 600 technically-oriented respondents, the worst problem of the Internet is stupid users [10]. Furthermore, stories about not so bright users can easily be found all around Internet from sites like Computer Stupidities[21]. Few examples from this site are presented below:

- A computer had been shut down and the user cannot turn it on. (The user was not aware that there is a power button).
- A user buys a device with a trackball and thinks it is a scanner. (The packet said 600 dots per inch tracking resolution).
- A user spends hours staring at a screen saver and then calls technical support and claims that the computer is broken. (Yes, the user did not touch the mouse or keyboard).

According to the well-known technology humanizer, Alan Cooper; "t*he number-one goal of all computer users is to not feel stupid*" [26]. Sometimes this goal might be difficult to reach, but by considering users and their UX as early as possible during the development it becomes easier. For example, according to Roto et al. [134] clearly stated UX requirements and defined target users considerably help to communicate and keep the focus during software development. Further on, by making it possible to use common sense and clearly showing important everyday controls users might not feel stupid. For instance, the *Greenhouse* reference game shows all of the objects that can be clicked in the screen, highlights them when they are selected, and shows a verbal explanation for the object in the place reserved for it. The game logic and functionality remains hidden behind the UI and the player only sees those objects (s)he can interact with.

3.7 Actors of software development project

If software causes difficult situation to users or clients, the question needs to be asked; who or what was the cause? And more importantly what could have been done during the software development project to prevent the difficulties? The answer is highly dependent on the size of the organization and the actors involved in the development project. Following subsections explain how different acting parties of software development project can affect to the outcome.

3.7.1 Managers

Management plays a key role in software development projects, yet their actions are not viewed without some misgivings:

20 http://www.1-famous-quotes.com/quote/33971
21 http://www.rinkworks.com/stupid/

- Finished product according to manager runs on target computer and doesn't crash. In case 'the finished product' is just a functional prototype, manager looks at it and asks why we cannot use that. The answer is too technical and filled with uncertainty to have sufficient force to convince a manager who sees an opportunity to save months of expensive effort. [26].

- Managers seldom, if ever will admit to making mistakes. They know all there is to know. It is the users and others that are the stupid ones [39].

Software development projects are very complex and multidimensional entities which are particularly susceptible to failures [13]. In fact, it has been shown that the job of the manager is one of the most difficult tasks in the software development project [61]. Still from the above statements, it is easy to get a picture of a manager who just sits in his or her office and puts pressure on the employees. These bad managers, who do not possess the required features or the will to learn, might have big influence on the outcome of the software development project.

Successful management of software projects is a complex and time-consuming job containing tasks like, organizing resources, scheduling, analysing risks, selecting methods and technologies, contacting clients, attending board meetings and dealing with budget issues [53]. Good management requires a strong manager with enough experience and good general mastery of the field of software development to keep everything in check [61]. Adequate knowledge about the whole software development process and its phases should be clear before the project even begins, regardless of the actual used design method or development model.

This dissertation work will not give direct benefits or tools for the managers of software development project. Indirectly it is possible that the increased UX knowledge among other actors will give them enough validity to reassure managers to consider UX more thoroughly. However, it finally depends on the manager if the proposals received from different acting parties of the development project are taken into consideration.

3.7.2 Designers

In this subsection, the term designer is not limited to any particular area of design but is used as a common term for all people in a software development project involved with design. The act of designing is essentially problem solving involving evaluation of different possibilities, the making choices, the use of definitions, and the making of trade-offs between different factors [19]. Löwgren and Stolterman [103] however, are of a different opinion. Problem solving, according to them, indicates that a problem is solvable and has a right and wrong answer. Design, on the other hand, needs to adapt to a changing and growing understanding of the situation, and therefore, it is difficult to determine if a design proposal is right or wrong [103].

According to Cooper [26], the design team should be the bridge between the programmers and end users. It is the job of designers to produce the ideas based on the wishes of the end users' and construct the character of the product by choosing and combining appropriate features [58]. The *SAP Design Guild* has published golden rules

for bad design [143], which everybody who participates in software development project should be familiar with. The idea of the golden rules is to encourage designers to do the opposite and thus avoid pitfalls. The list contains 18 rules, with reasoning and examples, which all have a major to minor impact on UX. The most relevant ones for the scope of this dissertation are presented below. These chosen rules have a direct influence on UX and might actually lead to a situation where a user stops using the application.

- Keep away from end users because you are the expert and know what users need. Since you know what they need, why should they need something else?

- Make it illogical because everybody knows that illogicality is no obstacle for professional users, but beginners will suffer.

- Use abbreviations wherever possible, particularly where there would be space enough for the complete term because abbreviations make your application look more professional.

Lack of knowledge in the field of design should not be a problem, but nevertheless numerous examples of bad or impractical designs exist. For example, an automated coffee machine shows three step instructions on the screen to get coffee from the machine. The instructions do not say anything about placing a cup in the machine so the user naturally assumes that the machine will dispense a cup. The coffee machine, in this case does not dispense a cup, meaning that the coffee ends up on the floor [12]. Another example of defective design comes from author's personal experience:

> Last year I was participating in a conference in Iceland, on the final day we had this wonderful excursion to a Blue Lagoon Geothermal Spa[22]. The place itself was absolutely fantastic and I will certainly take my family there sometime. But there was one slight problem, which at that time did not affect me directly, but many others from our party. Blue Lagoon has a nice bracelet system which could be used to buy beverages while in the pool and the same system is also used for the lockers. The idea is really nice, but the Blue Lagoon has around 20 dressing rooms, which all look the same. Each dressing room has around 50 lockers without numbers and every bracelet could be connected with any of the lockers as long as the locker was unoccupied. The problem was that there was no other way of checking which locker you have picked, but to go back to reception in your trunks or swimming suit.

Simple bracelet reader in every dressing room would have solved this problem. Even though, this example was not a software design problem, it demonstrates the importance of design.

Software development projects in big companies may include designers from multiple different areas, such as UX, UI, interaction and graphics, and the company might invest millions just in design. But lots of money does not necessarily lead to good outcomes. For example, in the past Microsoft invested millions in interface design, but its products were commonly disliked [26]. This can be partially explained by the rumored quote by

22 http://www.bluelagoon.com/Geothermal-spa/

Bill Gates about Windows 95 or Windows 98: "*If you can't make it good, at least make it look good*"[23]. Currently, Microsoft appears to have noticed that good UX has become a point of differentiation in the enterprise space and they now act accordingly [108]. In a small company, on the other hand, the opposite may be true, the same person might be responsible for every area of a project that has something to do with design. Regardless of how many people are working in design, in the end everything must work seamlessly together. It could be said that design is like a building a house: If the foundations are not right, no amount of decorating can fix the result. Also vice versa, superior design is useless, if it cannot be build.

Even if all available design knowledge and end user requirements are transferred successfully into product character and the product is perfectly constructed, there still is no guarantee that end users will perceive and appreciate the product in the way it was intended (see Moomin moped example, Figure 4). In 1990 Green [52] noted the problem that neither the expectations of designers nor theories of user behavior allowed for the natural human behavior of changing one's mind.

For the designers this dissertation and the implemented utility will offer a way to get some starting points for the UX design. Another possibility is to check the premises of existing designs or sketch against the sentiments of end users. Finally the prototyped *Wizard of UX* could aid designers to speak the "same language" with the programmers.

3.7.3 Programmers

"*Programmers are not evil. They work hard to make their software easy to use. Unfortunately, their frame of reference is themselves*" [26]. For most laypeople, the term programmer conjures up a vision of a nerd with long greasy hair sitting in a dark room, with the only light coming from the monitor. The IQ is close to 200, but social skills are comparable to a freezer. While such mythical beings exist, they are a very small minority. An average programmer is just like any other person, (s)he just happens to know how to read and write another language. According to Cooper [26], there are programming shops whose staff does not have a slightest clue about designing for end users. On the other hand, programmers have strong opinions about what they personally like, commonly those things that are easiest to implement. Therefore it is natural that when programmers must do design, conflicts easily arise between implementation and the needs of end users [26]. According to a study conducted by Vukelja et al. [165], programmers for example frequently develop user interfaces alone and this happens in nearly 50 % of all cases. Sometimes programmers are condemned to do design because there is no one else to do it, but it is not rare to encounter a programmer who thinks (s)he is capable of doing the design.

Despite the finger of blame commonly being pointed towards programmers if software fails, they should not be the only ones to be admonished. An example from the software development project of the *Activator* presents some issues that happened due to managers.

23 http://thinkexist.com/quotation/if_you_can-t_make_it_good-at_least_make_it_look/330034.html

> Whole software development process, excluding client requirements and deadlines, was under authors charge. Client requirements and deadlines for the *Activator* were set during negotiations by project leader co-operation with the client. The author was not there and could not provide information about what is possible and what is not. During the project, client changed and added requirements. The author was present, but was unable to convince client or project leader that the requirements were unrealistic within the given time.

If the above scenario would be transferred into business world the situation could be a lot worse. For example, according to company policy overtime work might not be allowed, but everything must be ready before the promised deadline. Designers are not willing to alter their designs if programmer cannot implement something. Strict deadlines drive programmer(s) to write code without comments and exhausted programmer might leave the company during the project. These conflicts might easily lead to a situation where the programmer has no other option, but to take control and decide something unilaterally. The problem is that the programmer is good at what (s)he is trained to do, which is programming, but seldom possesses enough awareness of UX or HCI to be able to make acceptable modifications to the design [26].

This dissertation will not transform programmers into designers, but the prototyped *Wizard of UX* will raise programmers' knowledge of UX and offers a view through end users eyes. Furthermore, this utility will aid programmers during the early phases of software development if they have been condemned to consider the UX aspect.

3.7.4 Testers and evaluators

Testing in a natural language usually involves act of trying different possibilities. In a software development project, testing means all actions which are used to measure and enhance the quality of the software. The chosen development method defines when and how thoroughly testing and evaluation during the software development project takes place. In addition, time, money and available equipment also define the amount of testing that can and will be done during the test phase(s) [53]. In an optimal situation, every possible situation would be tested, but in most cases this is not possible [115]. However, even this procedure would not prove the application to be completely defect-free. In a real world situation, it is impractical and virtually impossible to find all defects in an application, even trivial ones [115]. In fact, it is estimated that 5 % of all software defects remains undiscovered forever [53].

Many terms exist that describe a problem in an application, including; bug, fault, error, flaw, malfunction, defect, infection, failure, etc. In many cases, the terms are used almost as synonyms. In this dissertation the following terms, defined in [174], are used: Defect is an incorrect program code, infection is an incorrect program state and failure is an observable incorrect program behavior. The following sentence clarifies further the terms and their usage: The defect caused an infection, which led to a failure.

There is a great amount of knowledge in the field of software testing. For example Amazon book search on 13/05/11 found 76 books written after 1.1.2010 mentioning the

term "*software testing*". If Google Books is used, 1900 results are found. When the timeline is extended to start from 2000, Amazon finds 410 results and Google 27 600 results.

In spite of the knowledge, there are problem. For example a differing views of software problem. A clear infection from the end user point of view may be considered a feature from the software vendor point of view. Therefore, the specification should always contain detailed interpretations about all possible situations. Another problem is the experience of the tester conducting the tests, which have an influence on the results [115]. For instance, if a programmer without adequate experience in testing performs the test, his or her starting point could be something like; to demonstrate that there are no failures or to show that the application performs its intended functions correctly. Whereas a qualified and experienced tester will begin with the starting point: The intention of the testing is to produce infections and failures so that the defect(s) can be fixed. These two test cases begin with different starting points; one case tries to demonstrate that the application is failure free, whereas the other tries to produce failures. Furthermore, while testing can be conducted inside the company, evaluation should happen outside the company and includes either professional evaluators or test users. With outside test users, the Hawthorne effect is likely to happen and should be taken into account when the results are evaluated [76].

This dissertation work will not give tools for the testers or evaluators. However, the knowledge provided by this work and the *Wizard of UX* should help to testers focus on areas that were identified as important for end users.

3.8 Evaluation and analyses of UX in software development

Regardless of the precise software development process followed, the development broadly contains seven phases; requirements, definitions, design, implementation, integration, testing and maintenance. It should be noted that the phases may be named differently and may have different emphases. If the whole development is done by a single programmer, (s)he is unlikely to follow any process by the book. Nevertheless, each of the above-listed phases can, at least to some extent, be identified. Actors in the development project, used development model as well as design methods all have an influence to outcome of software development project. Since UX and software development are both huge multifaceted disciplines, their integration forms a challenge [79], [134], [166]. In addition to the challenge, it has been a long-term wish, especially from companies, that UX could be evaluated and analysed efficiently, especially during the early phases of software development [149].

In order to demonstrate why it is important to consider UX from the very beginning of software development, an estimate by Schach [144] is presented. In 2002 he estimated that life cycle costs of software are divided as illustrated in Figure 8. The large maintenance section consists of fixes, patches, new features and modifications. Ultimately most maintenance tasks can be traced back to the end users of software [53].

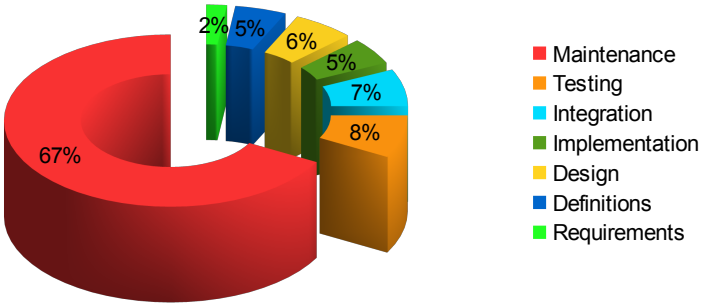


Figure 8: Division of software life cycle costs

According to Haikala and MäriJärvi [53], above figure is a typical example of the division of software life cycle costs. While the precise figures vary from source to source, it is not rare that maintenance accounts for over 50 % of overall costs [53]. Therefore, greater effort in ascertaining the requirements, definitions, design, and implementation phases could be expected to reduce the time and money used in the maintenance section. This dissertation offers an easy, practical way to put more effort into early phases of software development when UX is considered.

Vermeeren et al. [163] saw the need to identify the current state of UX evaluation methods and the characteristics and qualities of UX evaluation. They studied different methods used in academic and industrial environments by collecting examples from workshops, SIG (Special Interest Group) sessions, conferences, online surveys and literature. In addition, they also included results from earlier studies conducted by different instances. In total, their results contain 96 methods with analyses of strengths and weaknesses. Most these results are published on a web site *All About UX* (footnote 1 page 1), which at the time of writing offers the latest information and is the most thorough website dealing with the topic. The web site presents different ways to categorize UX evaluation methods, e.g. according to type of study field; lab, online, survey, or according to period of experience; before, during, after, long-term. The different evaluation methods may also have special requirements like expert knowledge in psychology or the ability to use complicated statistics software.

The focus of interest of this dissertation is UX during software development, so it is natural to use categorization based on development phases. The seven phases of software life cycle have been mapped into materials available for UX evaluation (scenarios and sketches, early prototypes, functional prototypes, products on market) presented *All About UX* website (footnote 1 page 1). Different UX evaluation methods suit different development phases. Many of the presented methods suit multiple development phases, but the suitability is biased to the latter end of the software life cycle. However, the sooner the UX problems are discovered, the better [149], [163]. Evidently the need for early phase UX utility exists.

Figure 9 extends the Figure 7 by presenting currently available UX evaluation methods for different development phases. The target of this dissertation, the area between the

first idea and methods suitable for scenarios and sketches, is highlighted with reddish colour. In the figure, the software development life cycle follows the center arrow from left to right, starting from the idea and ending with a product in active use that needs maintenance. Colours from red via orange to green, as well as the height of the rounded shapes gives a trend about the amount of methods in each category.

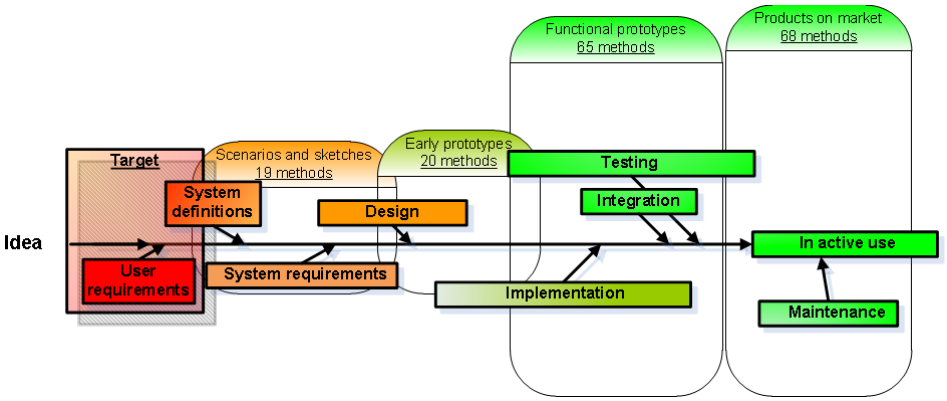


Figure 9: Software life cycle with materials available for UX evaluation and the target area

3.8.1 Scenarios and sketches

The idea forms one end of the target in Figure 9, and scenarios and sketches the other. Considering UX when user requirements, located inside this target area, are collected and evaluated, is the goal of this dissertation. Partially this dissertation reaches to system definitions and system requirements as well since those should be connected to user requirements. At the beginning of a development it is common that multiple competing ideas exist, but ideas cannot be interacted with. Since the scenarios and sketches phase is next to the target area, the suitability of conceivable methods under this category for evaluating the UX of idea and user requirements is analysed.

Methods that according to authors' judgment are not suitable for the scope of this dissertation are left out from the Table 1, which is formed by utilizing method descriptions presented in [6], as well as original sources when available. When original source is used, reference is mentioned after the method name in the first column. Table presents method name and gives the pros and cons for the method. This work contributes to the table by giving a brief suitability evaluation of the method for the target area when software development, conducted in smaller companies or by individual developer, is considered. The methods presented in highlighted rows are introduced more thoroughly later in the chapter, due to their being partially similar to solution introduced in this dissertation or appearing usable for idea or user requirement analyses.

Table 1: UX evaluation in scenarios and sketches phase of software life cycle

Method	Cons.	Pros.	Suitability for the target area
Co-discovery (Co-participation) [38]	*Participants must be carefully screened *More costly, laborious analyses due to 2x participants * Results dependent on learning, verbal and cultural aspects *Requires laboratory	*Produces authentic responses due to natural interaction style *Captures initial contact with the target *Easier for the experimenter (during the test)	*Useful for gathering user requirements from idea. *If discussions are ruled properly, also UX importance of user requirements could be collected *Might also be suitable for idea generation.
Contextual Laddering [173]	*Requires lots of effort and skilled interviewer *Laborious analyses	*Reveals why something is important *Reveals value chains	*Cannot be used for idea or user requirements generation *Could be used for evaluating existing user requirements, but too laborious for small development companies
Emofaces [32]	*Cannot separate different emotions from another *Requires something ready	*Fast to conduct *Nonverbal so culture independent	*Very usable and fast for evaluating already collected ideas or user requirements *Cannot be used for idea or user requirements generation
Kansei Engineering Software [145]	*Takes much time *Requires knowledge of psychology, statistics and engineering *Only available if joint project	*Used in industrial context. *Translates feelings into product properties	*Probably too laborious and unreachable for small development companies. *Could be usable for evaluating ideas and user requirements, if simpler and could be enhanced
Paired comparison	*Amount of comparisons grows too large if stimuli is large *Can only be used to compare according to one attribute at a time	*Easy to use, select best from pair(s) *Suitable for different settings	*Usable for evaluating already collected user requirements *Cannot be used for idea or user requirements generation
Playability heuristics [91]	*Only for evaluating games *Requires laboratory	*Fast and cheap since heuristics are freely available *Can be applied early	*Some heuristics could be used to evaluate already existing user requirements or ideas *Cannot be used for idea or user requirements generation
Private camera conversation [38]	*Requires laboratory *Requires some sort of prototype to play with *Participant may not feel comfortable to talk to a camera *Analyses takes time	*Can be combined with co-discovery *Might produce more authentic responses than face to face interview	*Usable for idea generating or user requirement gathering if there are enough participants *Setting up a lab and analysing videos might be too laborious for small development companies
Repertory Grid Technique [42]	*Requires lot of effort from experimenter and participant since individual RG table is built for each participant.	*Open and dynamic *Is qualitative and quantitative	*Usable for evaluating already collected ideas or user requirements *Cannot be used for idea or user requirements generation

Semi-structured experience interview [105]	*Requires experienced interviewer *Analyses takes time	*Fast to conduct since only small amount of participants needed *More flexible than standard structured interviews	*Could be used to generate ideas or user requirements by giving participant some basis
Sentence Completion	*Participants do not necessarily have time to fill every sentence *Effort taking to analyse sentences	*Participants can freely express themselves *Test are easy to administer	*Could be used for idea or user requirements generation with carefully chosen sentences
UX Expert evaluation [167]	*Experts are not real users *End users are not involved *UX experts are still scarce	*Fast and inexpensive	*Usable for idea and user requirements generation *Results should be compared with end users

Co-discovery [38]**,** is a method where two participants explore the given sketches and scenarios and talk about them with each other or with the moderator. The conversations between the participants are either recorded and supervisor is not present or supervisor is located at the same room as participants and takes notes [38]. This method can be further enhanced by utilizing friends who already know each other and communication is assumed to be natural and may trigger more experiential comments than discussions with a stranger or moderator. The positive aspects of this method are its authentic responses and its suitability for investigating initial contact with the task. The main weakness lies in difficulties controlling the direction of the conversation, especially in cases without a moderator [6]. Another weakness when considering the suitability of the method for the target area is the number of ideas and requirements from which the relevant ones needs to be sifted. Nevertheless, this method could be used to produce excellent new ideas and some requirements, although the participants must be given directions that are strict enough to keep them on track but at the same time loose enough to give them enough space to produce innovative ideas and practical requirements.

Emofaces [32], is a method developed to overcome the difficulty that emotions are not easily put into words or sentences. As its name suggests, it is based on drawings of facial expressions and the test person is told to pick one or more card that expresses his or her current feeling about the test subject. This method is fast to use and applicable to any culture. The main weakness is an inability to separate different emotions from each other [6]. For example, someone might express dislike with the same emoface in various cases, e.g., it was not aesthetic or the usability was bad. Therefore, some verbal extensions to emofaces are needed, which is commonly an invited verbal description of the emotional response. The method could be used to test already gathered ideas and requirement, but is unusable for idea or user requirements generation since there are only pre-defined images.

Kansei engineering software [86], is a method for translating feelings and impressions into product parameters [145]. This method is used by several companies, e.g. Electrolux, Saab and Skanska [86] and was developed when it was recognized that

companies often want to quantify customers' impressions of their products. Kansei Engineering can measure feelings and shows their correlation to certain product properties [145]. In consequence, products can be designed in a way, which corresponds to the intended feeling. It is based on predefined words in software, product properties and ratings given by users to the product in question. When all these aspects are combined, the method tries to predict an optimized layout for the product [145]. Approach to UX in software development described and prototyped in this dissertation, described in Chapter 5, utilizes a similar approach when important UX attributes for the target group are predicted and visualized. The Kansei engineering software method could be utilized in the target area, if enough pre-collected end user data from the field is available.

UX expert evaluation [167], is commonly used evaluation method in the area of software development. This evaluation can be conducted for instance with a cognitive walk-through, in which expert evaluators write down all positive and negative aspects affecting the UX of end user. UX expert evaluation method has been criticized for relying on users that are not the actual end users of the systems [6], and even UX professionals themselves admit [167] that it is challenging to try to see things with the eyes of other users. The positive aspect of this method is that, if no extended period of evaluation is required, which makes it fast to use and cheap, but still according to Väänänen-Vainio-Mattila and Wäljas [167] longer period of time might affect on findings since UX changes over time. In addition, UX expert evaluation method is usable in every phase of the development project, including the target area, and can be conducted online. Negative aspect of this method is that, the requirements and definitions would be based only on the viewpoint of experts, and some valuable views from end users might escape from the focus. This dissertation suggests a solution that makes it possible to consider end user opinions without possibility to actually meet them.

3.8.2 Functional prototypes and products on market

Methods inside the early prototypes group were left out from this dissertation since they all require something concrete to work with and utilization of these methods for idea generation or user requirements gathering would require modifications. From the functional prototypes and products on market groups, methods that are related to survey research presented in this dissertation were selected for further discussion. It should, however, be noted that these methods are not usable, unless modified, before at least a functional prototype is implemented.

The ServUX questionnaire [6], is a recently developed evaluation method. It covers a wide set of both pragmatic and hedonic UX aspects such as usability, UI consistency, functionality and reliability as pragmatic values, and aspects like privacy in the hedonic side. All of these attributes were included in the survey conducted in this dissertation.

AttrakDiff is similar a questionnaire that tries to assess the users' feelings about the system by studying both pragmatic and hedonic sides of UX. Currently, AttrakDiff has its own web site, where the basic version can be utilized free of charge. The basic version offers 21 predefined word-pairs which the respondents evaluate. In the 'pro'

version customers are able to define the word pairs [56]. The idea behind some word pairs is transformed into a new word, e.g. predictable – unpredictable is transformed into consistency of functionality, and practical – impractical is divided into usability and functionality in this dissertation.

Surveys and questionnaires have a long and successful history for gathering opinions and information from adult individuals and are apparently a common way to collect information also about UX. Hsu et al. [63], for example used a questionnaire to measure the gaming experience of college students with eleven factors to understand addictions to MMORPG (Massively Multiplayer Online Role Playing Games) and Bargas-Avila et al. [14] used a questionnaire to measure user satisfaction with an Intranet. Strength of survey approaches is the possibility to conduct online or easy transformation into online format. Furthermore, online survey tools offer a possibility to get the collected data, in e.g. Excel format, allowing easy basic analyses. Only validation of data and outlining needs has to be done by the moderator. Despite being successfully and frequently used, surveys can be criticized: When for example a UX survey is conducted after the use case, it only assesses the recollection of experience, not the experience during use. However, evidence that explains the real difference between recollection of experience and actual use experience was not found.

3.9 Discussion

During the early phases of software development the idea or ideas are further developed into requirements that fulfill the needs of users. Unfortunately, current software development life cycle models do not pay sufficient attention to designing for UX, instead it is assumed to be built into the model. E.g. in a Scrum model, considering UX belongs to the Scrum team, but there are no rules how this should be done [157], it is up to the team to decide.

There exist nearly hundred methods that can be utilized for evaluating and analysing UX during a software development project as the list [6] presents. However, majority of these methods are usable in the latter phases of software development, and a scarcity of methods exist in the early phases. The problem is often exacerbated since some developers and development companies have little knowledge about taking UX of end users into account [26]. This context emphasizes the validity of the research objective O1 (*Raising the UX knowledge among software developers.)* and O2 *(Considering the attributes of UX by utilizing statistical user profiles in the early phases of software development.)*. The first objective is self-evident, when general knowledge towards certain aspect is raised; also the ability to take it into account is raised. The second objective, in spite being suitable for complete software development life cycle, is most effective when utilized in the target area.

To conclude, every presented UX evaluation method has its strengths and weaknesses and compromises must be made between the wanted results and the resources available. This chapter forms the basis for the *Wizard of UX,* which aims to fill the research objectives.

4 Surveys: Collating and analysing UX knowledge

This chapter introduces four different surveys, the basis for conducted surveys, chosen respondents, and justifications for the different demographics and abilities included in the surveys. The latter part of the chapter presents explanations for the chosen UX definitions and included UX attributes, as well as results from the analyses of survey data.

4.1 Four surveys

The first survey (Baseline survey) [96] formed the basis for surveys conducted for this dissertation. The second and third were conducted as part of this dissertation (**UX professional survey,** end of 2009 and **End user survey,** beginning of 2010), (see Appendix I). The preliminary results of these surveys were presented in a UX workshop I-UxSED 2010 [82]. The fourth and final survey included was conducted under authors' supervision (Upper secondary survey, see Appendix II). Figure 10 presents the chronological order of the introduced four surveys. In the middle of the timeline, the surveys conducted for this dissertation are presented.

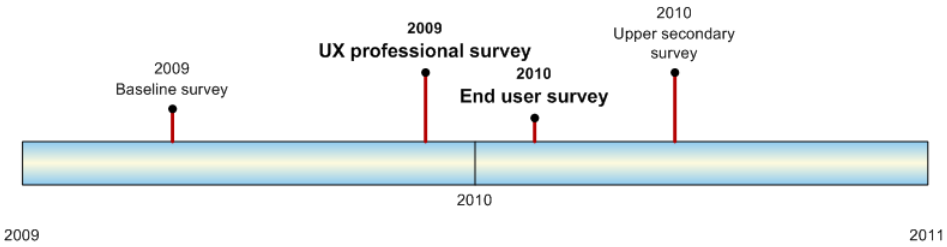


Figure 10: Survey time line

The baseline survey was conducted by Law et al. in 2009, and the results were published in the CHI 2009 conference [96]. The main objective of their study was to clarify the concept of UX with definitions and multiple statements of UX. The upper secondary survey was conducted in 2010 by a bachelor level worker, Ville Sallinen whose work the author supervised. The main aim was to establish how understanding of UX varies between university and upper secondary students. The aim in both of the surveys conducted for this dissertation, was to explicitly capture personal opinions about what is important in UX for a particular respondent and to indicate the effects of

demographic variables and ability to use computers. The thread behind this approach is to form an average importance based on multiple respondents with same or similar demographics and abilities.

Table 2 presents the main characteristics and differences between the four surveys and highlights the ones conducted for this dissertation. Computer ability in both highlighted surveys and upper secondary survey was divided from 1 to 6, where 1 is very poor and 6 is rock solid professional. Section 4.3 explains the background questions more thoroughly.

Table 2: Differences between surveys and respondents

	Baseline survey [96]	UX professional survey	End user survey	Upper secondary survey
Gender	82 females, 137 males, 56 unknown	9 females, 3 males	559 females, 801 males 5 unknown	42 females, 13 males
Age	18-59	25-59	18-64	16-19
Computer ability	NA	3-5	1-6	1-5
Respondents	UX professionals from academia, and industry	UX professionals from academia	University students from multiple different disciplines	Upper secondary students from Lappeenranta
Finnish respondents	18 %	50 %	95 %	100 %
Definitions	Five definitions	Six definitions	Six definitions + do not know / do not care	Eight definitions translated from English + do not know / do not care
UX attributes	NA	20, increased to 21	21	27
Other	Clause formatted UX statements			Conducted in Finnish

Results from the baseline survey are based on 275 UX professional respondents in academia and industry. From the Upper secondary survey, 59 responses were received, but four responses had to be removed from the analyses due to inappropriate values. For instance a row of empty values and increasing series from one to six were removed. Final respondents from the upper secondary survey include 13 males and 42 females aged 16 to 19 and all respondents were Finnish. Furthermore, this study was conducted in Finnish, so the results are not fully comparable.

UX professional survey was submitted to 20 UX professionals from which 12 answered. As the UX professional survey also worked as validation for the end user survey, the structure of both these surveys is the same; background questions, UX definitions, and UX attribute evaluation. Background questions referred to gender, age, nationality and

ability to use computers. The UX definition part was included to be able to demonstrate the divergence of understanding of UX between UX professionals and end users. The last part of the survey listed UX attributes. In addition, the UX professional survey contained a possibility to suggest new UX attributes giving their importance, as well as a possibility to give general feedback about the survey and importance of measuring UX.

End user survey, with a covering email, was sent to approximately 15000 students. This number of possible respondents included all registered bachelor-, and master level students from Lappeenranta University of Technology and University of Oulu. The respondents covered multiple disciplines like chemistry, IT, medicine, humanities, economics and business administration. It might be argued that this group of end users is too uniform to be generalized since they are all university students; this criticism has some validity. Of the current end user respondents, most of them (72%) are from 20 to 29 years old and 78% of end users have 3-4 level ability in computer use. Many different nationalities can be found from the responses, but 95% of all end user respondents are Finnish, so the effect of nationality is unlikely to be significant. Nevertheless, the results gained offer a decent overview of how different demographics affect appreciation of UX attributes. Naturally, it is possible to enlarge the group of respondents to cover a wider area of the population, but this has been placed into future work.

Answers to the end user survey were received from 1440 respondents, which is about 10 % of people who, in theory, received the survey link. When it is taken into account that students in Finnish universities are asked to fill a survey after every course, this percentage value is well in line with normal response rates. The received responses were checked for rationality and coherence utilizing statistical analyses. The complete response line was removed if the sum of the 21 UX attributes evaluated was outside the 22-110 boundary, in which the lower limit would mean an average of 1 (very important in every attribute) and the upper limit would mean an average of 5.23 (unimportant = 5 and do not know = 6). Another reason for removal of a whole response line is a standard deviation of zero, which indicates that the same value was given to every attribute. Response lines that contained an increasing or decreasing chain of values were also removed from the results. Once rationality and coherence checking had been completed, 1365 end user responses were accepted for final analyses. Finally, it was noticed that the gained results were divided approximately based on normal distribution forming a left side emphasized bell curve. Combination of UX professional survey and end user survey is presented in Appendix I.

4.2 Validating survey(s)

A two-step validation procedure was followed for end user survey conducted for this dissertation. The first step of validation produced the UX professional survey, that in addition to being the second validation step, also worked as a qualitative data producer.

1. The first validation step was a small-scale test and the intention was to discover flaws and find out possible ways to improve the survey. This test was carried out by submitting the survey to personnel inside our research laboratory. The test

revealed some technical problems with the survey and improvements were also made to the appearance, colours and arrangement of the questions.

2. The second validation step was content validation, which also acted as the UX professional survey. This second step was used to validate the questions and attributes included in the end user survey, but also to produce qualitative UX information for analyses. The survey was launched to 20 hand-picked UX professionals who in addition to answering the survey also had a possibility to give free feedback and criticism. Based on suggestion from UX professionals, questions were adjusted to make them clearer to avoid some of the misunderstandings, and 'do not know' / 'do not understand' alternative was added to the survey. Also the 'ease of taking into use' was added to the list of UX attributes. UX professionals were selected by browsing IEEE and ACM digital libraries and using Google Scholar with the keyword ‘user experience’ to find authors and co-authors of relevant papers. Naturally, other UX professionals exist and their number increases all the time, but 20 possible respondents was considered to be enough to produce sufficient amount of information for validation and analyses.

4.2.1 Feedback and criticism

In addition to the qualitative data received from the UX professional survey some feedback and criticism were received. Following four worthy comments were received from the respondents of UX professional survey:

1. “*Many different aspects have influences on user experience at the same time. Therefore, in order to evaluate experiences it is important to restrict the focus what you are studying at the certain case. Otherwise, the analysis and interpretation may be difficult to perform. Also, to find the proper research methods for studying user experience is the key element for successful studies. It is import is to enable that you can be sure what a user is experiencing and why.*”

2. “*With regards to this questionnaire, there should have been an option in section 7 called "do not know or do not understand". The questions were hard to evaluate as the answers are so context dependent.*”

3. “*It is unclear if we should think about ourselves as users, or rate the importance of the different factors for UX in general. I now rated the importance according to my personal values "to *you* when using sw", so it does NOT answer to what is important in UX in general.*”

4. “*I guess question 6 and 7 are about the same product/service? This is however unclear and the answers you get might not be representative or related to just one software product. Would it not be interesting to know what the software product one is thinking of is as well? I for sure would like to know :)*”

The first comment is a general comment about UX, but it contains a good point; the need to focus on a certain case. The fourth comment is related to the same issue. In light

of these comments, a question about the software in mind was added to the end user survey. The third comment is somewhat surprising. UX professionals were asked to "*Evaluate the importance of the following UX to you when using a software*". Intention was to collect professional evaluation of UX attributes from personal viewpoint and form the average general importance based on these evaluations. Naturally, it would have been a different thing if UX professionals had been given a target group that they should think about when performing the evaluation, but in this survey UX professionals were the target group.

The improved and validated end user survey was launched to university students conducting their basic studies in Finland during February 2010. The intention was to collect quantitative knowledge from a reasonably-sized group of end users. Viewpoints were captured from two different universities, Lappeenranta University of Technology and University of Oulu. Comments and criticism were also received from this survey. The most common comment and the most interesting criticism are listed below.

- *"What is UX / I do not understand UX."*.
- "*Nerds could try to use terms that anyone is able to understand. Why on purpose make things more difficult than they actually are. Absolutely a waste of time. If an intention is user friendliness, I can only state that we are light years away from the goal. We do not have a common language that we could use to find out what is wrong with usability!!! Computers and programs are designed to be useless, thus the main goal of people working in the field is using such terms that ordinary people cannot understand*.".

The upper comment was nearly always related to a line that was removed from the final analyses. After receiving similar comments, the UX abbreviation was opened, and the number of such comments was reduced, but not completely, since the term user experience was also found to be too difficult by some respondents. Verbal comments such as the lower comment were most valuable since they reveal what end users think. The comment is a translation from Finnish to English, but hopefully captures the frenzy and anguish of the original. The comment reinforces the whole intention behind this dissertation; to consider end users with different demographics and abilities.

4.3 Background questions

This category of questions is used to collect personal demographics variables of respondents in order to be able to categorize answers during the analyses and contains gender, age, nationality and ability to use computers. Gender is commonly studied in the field of social sciences and psychology, and multiple studies, e.g. [112],[113],[114] have shown gender to be important in, e.g. product branding, aesthetic opinions and design preferences. Rosen [133] and Eriksson and Lindholm [40] have shown gender to have impact on abilities like reading and math performance. In [35] Diederich states that female students in particular may need academic counseling and that their career plans require more flexibility than their male counterparts. Obviously, gender matters. Therefore, it is justified to have gender as one of personal demographics questions.

Age has also been considered to be important; Sykorova, for example, shows age to be a relevant component of an individual's identity and a factor in many contexts [156]. She writes about age denial and how perspectives towards life changes with age. Interaction between technology and elderly people has also been studied, e.g. special requirements for using a mouse [142] and individual rehabilitation and well-being [41]. In a recent study by Yildirim et al. [172] the authors show age and gender to be an affecting factor when studying child-computer interaction, which is at the other end of the 'user scale'. Read concludes that psychologists and educators suggest that for children, questions should be asked in a positive way [131], which is completely different from the adult world where questions should be placed in a way that they will not raise either positive or negative feelings. The divisions used by Statistics Finland[24] to categorize age have been utilized.

Ability to use computers is also considered in the demographics questions, since HCI is directly connected to the ability to use computer technology. To be able to ensure that all respondents understood the question similarly, possible responses were tied to the scale given below. Respondents did not see the number, only the verbal definition. The numbers presented here are used in the analyses and charts.

1. Very poor.
2. I can use basic programs.
3. I can use, install and update programs.
4. I can develop / maintain minor programs, web sites, etc.
5. I can develop / maintain advanced programs, web sites, etc.
6. I consider myself as a rock solid professional.

4.4 UX definitions

UX definitions are not included in the *Wizard of UX*, but this part was included in the surveys to demonstrate the divergence of understanding of UX between UX professionals and end users. Many definitions for UX exist in literature, for instance [6] has gathered 27 different definitions found from different sources. The work in this dissertation was conducted before the launch of the site, so the included definitions were collected from elsewhere. The intention of this part was to partially reproduce the baseline survey, although some modifications were necessary.

In the baseline survey, participants were asked to select the most preferable definition out of five definitions (d3, d4, d5, d6, d8). According to the authors of the baseline survey, these definitions were chosen, since they represent many perspectives from multiple different instants from academia and industry. It was decided to omit definition d8 since it was considered too closely related to a company and its services and products. Definition d6 was removed due to its low support in the baseline survey.

24 http://www.stat.fi/index_en.html

Definitions d1, d2 and d7 on the other hand were included in the survey, since they were considered suitable.

- *d1 = All aspects of the user's experience when interacting with the product, service, environment or facility.* [70]
- *d2 = User experience is a special case of experience, where the person can use a system, with or without a purpose. Using means that the user not only senses the system, but also has the opportunity to manipulate or control the system* [139]
- *d3 = UX is a consequence of a user's internal state (predispositions, expectations, needs, motivation, mood, etc.), the characteristics of the designed system (e.g. complexity, purpose, usability, functionality, etc.) and the context (or the environment) within which the interaction occurs (e.g. organizational / social setting, meaningfulness of the activity, voluntariness of use, etc.)* [55]
- *d4 = The entire set of effects that is elicited by the interaction between a user and a product, including the degree to which all our senses are gratified (aesthetic experience), the meanings we attach to the product (experience of meaning), and the feelings and emotions that are elicited (emotional experience)* [31]
- *d5 = The quality of experience a person has when interacting with a specific design* [162]
- d6 = The value derived from interaction(s) [or anticipated interaction(s)] with a product service and the supporting cast in the context of use [168]
- *d7 = A momentary, primarily evaluative feeling (good-bad) while interacting with a product or service* [59]
- d8 = All aspects of the end-user's interaction with the company. Its services and its products [122]

Definitions written with *italic font* were included in both of the conducted surveys. The biggest difference to the baseline survey is that in surveys conducted for this dissertation the respondents were asked to select the three most suitable definitions for UX and mark the best with 1, second with 2 and third with 3. It was considered that this three point survey arrangement will give more thorough information about the mutual order of the definitions among professionals and end users.

4.4.1 UX definitions - results

Thorough evaluations for UX definitions are not given due just twelve UX professional respondents, which is not adequate to gain statistical validity. However the results are still presented in Chart 1 to be able to demonstrate the apparent trend that; an average end user of software has little interest in UX definitions. Responses under the dc column were divided as follows: "*I do not care*" – 26 % and "*I do not know*" – 9 %. In other words, a quarter of the possible end users of software seem to have little regard for UX definitions. If the dc combination is omitted from the analyses, then end users seem to

support d1 and d4 somewhat more than the other definitions, but every definition received a fair amount of support.

Among twelve UX professional respondents, definitions d3 and d4 are the most popular. The result is not statistically valid, but still it gives a similar trend than the baseline survey [96], in which most of the support were also given to definitions d3 (31%) and d4 (21%).

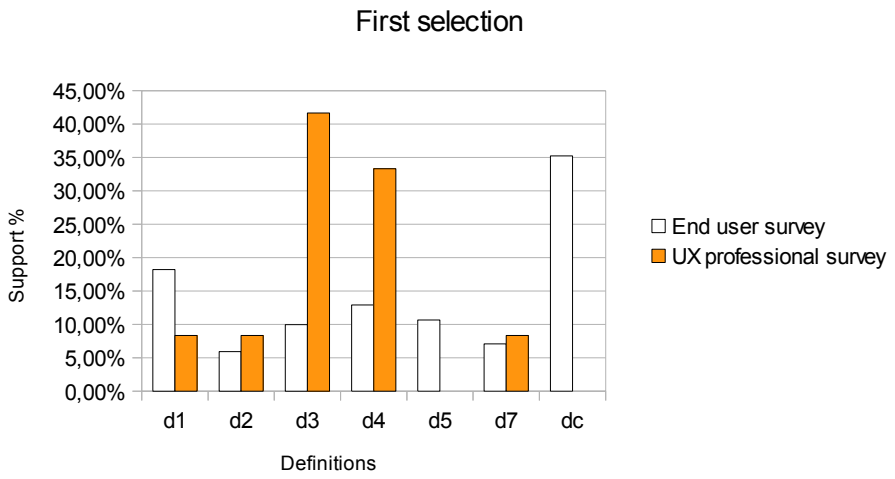


Chart 1: UX definitions support % by different respondents.
Note! Only 12 UX professional respondents

The UX definition part was also included in the upper secondary student survey, but its result cannot be included as the translation of the definitions into Finnish radically altered the actual meanings. However, most support (32 %) in this survey was given to the definition; "*User experience covers all phases of using a product from opening the package to daily usage as well as maintaining the product.*" that was not included in any other of the surveys. These differences in responses further indicate the subjective nature of UX.

4.5 UX attributes

To be able to demonstrate more thoroughly the dissimilarities between end users with different demographics and abilities UX attributes are evaluated further. This section is the primary part of the surveys conducted for this dissertation, and produced the data for the proof of concept application; the *Wizard of UX*. The received data was checked for rationality and coherence as described in section 4.1 and was used to build a UX database utilized by the *Wizard of UX*.

In science the term attribute refers to a characteristic of an object or entity, so an individual UX attribute can be seen as a one character of user experience. The UX attribute list is far from complete, but was considered to be comprehensive enough for studying the effect of gender, age and ability to use computers in UX. Furthermore, it

should be noted that care needs to be taken, on the one hand, not to lose important information, but on the other hand, not to generate an overwhelming amount of data.

The presented UX attributes are not based on any particular theory but are collated from various books, scientific publications and web sites. The original sources for the UX attributes included cannot be given, since many attributes can be found from multiple sources and establishing original references would have been extremely laborious or even impossible. Definition of attribute given after the name is taken from the Merriam-Webster online dictionary [107] when available. It needs to be clarified that these definitions were not given to the survey respondents since the intention was to capture the first reaction towards the attribute. Following subsections explain the included UX attributes.

4.5.1 Accessibility

"Capable of being reached.", Capable of being used or seen."[107]. Accessibility may also mean accessibility of information and knowledge [11]. It can also mean for example different ways to access menus, a possibility to enlarge fonts or use text-to-speech, and ways to make the application or device more accessible for users with disabilities. Therefore, accessibility is here related to using something and is therefore strongly pragmatic and temporal. Accessibility is an attribute that is commonly debated when considering UX. A software development projects in which the author participated as a developer included people with MS-disease. The symptoms of MS vary considerably but vision and understanding problems, and problems with everyday activities are common. During the project, the importance of accessibility as a UX attribute became abundantly clear. Naturally, accessibility is not such a pertinent issue for users without disabilities, but the importance of issues of accessibility has overwhelming support in the research community. Many industrialized countries have human-rights or disability-discrimination laws that make it a legal obligation to provide accessible websites [24], i.e. in USA it is named as The Rehabilitation Act and its amendments. In the light of the above, accessibility clearly had to be included in survey.

4.5.2 Aesthetics

"A pleasing appearance or effect." [107]. Imagine that an interior designer has just painted your living room wall bright red. At first, it may probably look dreadful, but after some time passes, your eyes get used to it and it no longer looks that ghastly. Similar effects might happen with software as well; after some usage experience, what might have seemed aesthetically bad at first is no longer so distressing. Theory has shown that aesthetics in particular have a critical effect on, e.g., usability, satisfaction, and pleasure [111]. According to [28] aesthetics are among the most prominent aspects of experience. However, aesthetics might be more important for some people than others but it is unclear for whom. To answer this question, aesthetics is included in this user demographics enhanced survey. It is possible to subdivide aesthetics. For example, Lavie and Tractinsky [95] have divided aesthetics into classical aesthetics and expressive aesthetics, in which the former means orderly and clear design and the latter the dimensions reflected by the designer like creative, special effect, etc. They state that these are clearly distinguishable from each other, but in the interest of simplicity and to

avoid introducing additional factors which might confuse end user respondents, this division was not used.

4.5.3 Brand

"One having a well-known and usually highly regarded or marketable name." [107]. Brand is a long-term, holistic, and hedonic UX attribute. It takes time to build brand image, which can be considered as the identity that the company reflects outwards. Brand may also be used for signaling trust and reliability [28]. One of the most recognized brands worldwide is Coca-Cola, which is known for its logo, but in general brand can be anything from a name, image or symbol to even a combination of colours in a certain order. Some researchers have mentioned the effect of brand on UX, e.g., Roto in [139] and [137]. Also Angeli et al. [28] have studied the importance of brand on consumer attitudes towards websites, but could not present any conclusive evidence. In Finland the Mannerheim League for Child Welfare[25] conducted an IRC (Internet Relay Chat)-gallery survey among youth from families of limited means. These youths revealed that brand and make of phone, shoes or clothes is an important aspect of personal image and, of course, a big reason for mocking those whose family cannot afford such luxuries [62]. So, in certain cases and with certain personalities, brand obviously has a great effect and is strongly dependent on the person, his or her demographic variables, and the effects of social pressure. Therefore brand was included in the surveys.

4.5.4 Consistency of functionality

"Agreement or harmony of parts or features to one another or a whole." [107]. Consistency is pragmatic UX attribute. By itself it means that contradictions do not exist and that there is a no dissonance between different parts or features, e.g. the order of the OK and cancel buttons is always the same and a button with the same text always performs the same action. In conducted surveys this attribute was named as consistency of functionality, in order to rule out e.g. firmness, density or resistance to movement. In theory of software testing, the 'comparing within the software' method utilizes this attribute and aims to assess if a feature works similarly throughout the software [71]. People do not want to use time learning things and they expect to be able to use learnt knowledge in many places. Just like traffic lights, which almost all around the world follow the same pattern, green-yellow-red. Therefore, consistency can be considered as a long-term UX attribute. Nevertheless, it needs to be borne in mind that, e.g., global consistency and local optimization can be at odds; thus, optimizing one thing may well cause problems with something else [17].

4.5.5 Coolness

"The state of being cool, just a higher standard"[26]. Coolness can be considered as the cliché part of the 'wow!' effect [64]. Coolness is based on many things; a combination of feelings, pricing, marketing, brand, aesthetics, etc. No simple explanation for coolness or being cool can be given, thus the terms are strongly holistic and hedonic, changes

25 http://www.mll.fi/en/
26 http://www.urbandictionary.com/define.php?term=coolness

over time, and are affected by social factors. For example, patting the head of a statue to get a 'good' photograph might be a cool thing for a tourist to do, but if this action is done in Thailand it is considered offensive by locals. Even worse, if the statue happened to be a Buddha statue, the tourist might find themselves in jail. In a software case, coolness might be the result of a new feature not seen elsewhere, stunning graphics or sounds, or even superiority compared to previous versions of the same software. Despite coolness being so multifaceted, opinions are undivided about coolness and its importance. For example, researchers have shown that youths are frequently attracted to peers who they consider as cool [16].

4.5.6 Device

"*A piece of equipment or a mechanism designed to serve a special purpose or perform a special function.*" [107]. A device can be, for example, a desktop computer, netbook or mobile phone. In general, all applications that can be launched on a mobile device are also runnable on devices which are bigger and more powerful, but not vice versa due to limited amount of memory or processing power. Angry Birds can be run with a desktop computer, but Photoshop is unlikely to be used with a mobile device. Devices can be touched, but still the device itself is only a dummy covering and an instrument for interacting with the functionality (embedded and installed software) underneath. In theory a device is a 'simple' attribute, but it is affected by many aspects like physical size, screen size, placement of functional keys, processing power, battery life, etc. For example, two 20-year-old males with similar interests might evaluate a device uniformly, but if one male is replaced by a 60–year-old female, then the results might be very different. Issues related to image and bragging, described in the brand section, also need to be kept in mind when considering the device attribute.

4.5.7 Ease of taking into use

Originally, this attribute was not part of the survey but was included following the validation tests of UX experts and their suggestions. Published definition for the term was not found, but ease of taking into use is clearly a short-term UX attribute. When this attribute is considered it needs to be kept in mind that what is easy for one person might be impossible for someone else. In general, this attribute means how easy it is to start using the new object, whether it is a phone, software or a whole computer. In addition to making things easy to start and to use, all possible steps should be automated but with a possibility to change things manually. For example, in MS Windows, an application bundled with Windows Installer offers pre-made selections but with a possibility for advanced users to change things manually. Samsung advertises its ES-series cameras by stating that it combines a number of easy-to-use features that will allow anyone to start taking digital images with ease right out of the box. Contrasting examples also exist:

> My USB (Universal Serial Bus) WLAN (Wireless Local Area Network) adapter installed fine and works fine under Windows 7. The chipset is not officially supporter by Linux Mint, the distribution that I use from time to time. The retail package of the adapter contained a driver for Linux and a statement that the adapter is compatible with Linux. This statement is factually correct, but the installation

instructions contained tasks like disabling the current driver from the kernel, adding things to a source code, compiling the source, etc. On reading the text, the average computer user would have forgotten the whole thing right away. Despite my knowledge of IT and experience as a developer, I tried and failed to make the necessary changes, mainly because of an incorrect kernel version and the fact that the instructions did not contain any hints about what should be changed if the kernel version is not the same as described in the instructions. Therefore, I decided that it is wiser and a lot easier to forget about the WLAN and just to plug a cable between the computer and router whenever I need to access the Internet from Mint.

4.5.8 Emotions

"A conscious mental reaction subjectively experienced as strong feeling usually directed toward a specific object and typically accompanied by physiological and behavioral changes in the body." [107]. Emotions are based on the limbic system that is responsible for non-rational thinking [7]. This makes emotions very subjective, hedonic, holistic, and tied to the current moment. Designing for emotions is, therefore, really challenging. However, some products, especially games, rely on evoking emotions in players. One such game is Final Fantasy VII[27], which is one of the very few games that have provoked emotions in me. Over ten years later the author still remembers certain parts of the game, especially where Sephiroth impales Aeris with his sword. User emotions are generally considered important in software or product development, but emotions alone are not enough. There is a need to know what is or was the cause behind a particular emotion [136]. Establishing this information is laborious and sometimes even impossible without all sorts of technical surveillance. Surveillance on the other hand may affect results, either because the user becomes restless and behaves unnaturally or because of the Hawthorne effect [76].

4.5.9 Environment

"*The circumstances, objects, or conditions by which one is surrounded.*" [107]. Environment is commonly referred to context, but it was assumed that the term environment is easier to understand for an average end user. Context, or in this case environment, is linked to social, physical and technical aspects [80], including e.g. weather, illumination, nearby people, location, time of day, etc. Environment is a holistic UX attribute and tied to the current moment. It is impossible for developers to effectively control the environment in which a device, application, etc. is going to be used. The importance of environment for users with variable demographics can, however, be studied. Alakärppä et al. [3] have studied pain monitoring systems in hospital and home environments. They found that pain monitoring systems have potential for home environment usage by patients, but a lack of resources in nursing personnel somewhat reduced the meaning of the device. Also, extra burdens, e.g. recharging, were seen as awkward at home. They also discuss that to be reliable, UX study requires observation in a variety of user environments, although they note that very often this is impractical or impossible [3]. Naturally, predictions about the most common places of use could be done but the object at hand should nevertheless be adaptive to changes in environment, e.g., automatic increase in contrast if the level of

27 http://www.ffonline.com/ff7/

luminous lux is high, the possibility to turn off sounds, e.g. while traveling with public transportation, or with mobile telecommunications devices the possibility to deny usage of mobile data connections when near national borders to prevent costly roaming.

4.5.10 Expectations

Anticipation: "*A prior action that takes into account or forestalls a later action.*" [107]. Expectations refer to the anticipation of user towards the application or software. The term anticipation would have been more suitable, but the term expectations was considered to be easier for an average end user. Expectation can be based on e.g. rumors, previous experience, advertisements or even just a hunch. Furthermore, according to Mäkelä and Fulton Suri [116], previous experiences and expectations have effect on experiences and create new expectations. Expectations are, highly hedonic, holistic and are formed cumulatively over time. Regardless of the origin of the expectations, many software development projects fail because the system does not meet the expectations of customers or users. Documenting customer requirements and getting agreement from them that requirements are fine is a common way for trying to avoid problems, but this approach relies on correct design and implementation. By utilizing this approach, a company is protected from many complains since the customer has signed an agreement of requirements. A common problem is that customers may not know precisely what they want, or they cannot express it a rational form, analogous with the case of an interior designer and customer. The customer might want the room to feel more spacious and moody, but she cannot give details; it is up to the designer to make loosely expressed wishes reality. Similarly, in software development, the requirements are often too loose to be useful for establishing if expectations were fulfilled or not [99]. A principle of least astonishment is a good way to start in the world of software development. Briefly, this principle states that software should not surprise the user in a bad way. Consider a situation in which someone is typing a password, e.g., for the auction site, Ebay[28]. At the same moment, the IM (instant messaging) client pops up and grabs the key strokes. The user is too excited about the auction item to notice the keyboard has been grabbed and ends up sending his/her password to an IM friend. This behavior is unexpected and most certainly unwanted. The surprised user is next likely to first go through the settings and then Google the problem and find out that the grabbing behavior cannot be changed, which might even lead to the IM application being abandoned.

4.5.11 Functionality

"The set of functions or capabilities associated with computer software or hardware or an electronic device." [107]. Functionality is a pragmatic UX attribute. According to Jordan, "*product will be useless if it does not contain appropriate functionality"* [77]. Despite functionality primarily being strongly pragmatic, it also has hedonic aspects which are dependent on user demographics and abilities. For example, it has been stated that enhanced look and feel influence functionality [146]. For instance, in word processing packages, for some users it is enough functionality that one can write, open and save documents, while for advanced users this basic functionality may be

28 http://www.ebay.com/

unsatisfactory since they are interested in e.g. special characters, a thesaurus and professional paging. While basic functionality, like, minimizing, maximizing, close, help, etc., should be offered without question, other features should be based on need(s) from the customers, clients, or end users. In the surveys conducted for this dissertation, functionality means functional features of an application, software or product that are available to the user.

4.5.12 Interaction

"*Mutual or reciprocal action or influence*" [107]. The meaning of the term interaction depends greatly on the situation. The field of interaction design is predominantly dedicated to designing meaningful relationships between users and products [89], but interaction itself might also be understood as social interaction between two or more users. In this work, both of these views were combined into interaction. Reason for this combination is that the two alternatives are fairly close to each other and with some modern technology it is sometimes even difficult to tell if the interaction is with a human or machine. The effect of different modes of interaction has been extensively studied, e.g. Nauman et al. [118] have studied multimodal interaction and its suitability for elderly people. Many game console manufacturers have also studied different methods of interaction (Nintendo Wii, Xbox Kinect, PlayStation Move, etc.).

4.5.13 Motivation

"A motivating force, stimulus, or influence." [107]. Motivation is hedonic and temporal and can be divided into intrinsic and extrinsic motivation [141]. An example of intrinsic motivation might be going to the gym to practice in order to bench press 100 kg someday. The intrinsic form of motivation is based on a will, interest or enjoyment that comes from the individual itself. Extrinsic motivation on the other hand is based on outside pressure. In a study conducted by Alakärppä et al. [3] nursing personnel were willing to use a recently developed pain monitoring system, if it supported their work and if the resources utilized could be regained one way or another. Obviously, at that time they did not have any intrinsic motivation for using the system. It would have been interesting to hear about their motivation after adoption of the system, as the theory of self-determination by Deci and Ryan [29] claims that it is possible to internalize extrinsic motivation if the task, work, etc. supports the values and beliefs of the user and therefore helps to fulfill the user's basic psychological needs [29]. Motivation from the individual itself is naturally the preferred form, since there is no outside pressure or affecting force. Intrinsic motivation is subjective. In addition, both forms of motivation are dependent on current mood, which again is affected by many factors [141]; therefore motivation also has great effect on other UX attributes.

4.5.14 Pleasure

"State of gratification.", "A source of delight or joy." [107]. Pleasure is a mental state of the human mind. It is claimed that the lack of joy is a new plague, a silent killer, which in the USA is the cause of a quarter of all health complains [128]. Despite being highly subjective, like other emotions, pleasure is associated with physiological and basic needs, as illustrated by Maslow [51]. Still the hedonic and holistic nature of

pleasure makes it extremely challenging to design for pleasure or evaluate it accurately, although attempts have been made; pleasure is the core of experience design and e.g., Chou and Hsiao [23] measured pleasure experiences gained from internet usage. Despite the difficulties in designing and evaluating pleasure, it is seen as a very important aspect for well-being and longer, happier life [128].

4.5.15 Price

"*The quantity of one thing that is exchanged or demanded in barter or sale for another.*" [107]. From the user point of view, price is the amount of money or other payments users have to use in order to get the object. From the company point of view, price might be for example the ROI of UX. When included UX attributes were considered, it was assumed that age may be a highly influential demographic factor when evaluating the importance of the price attribute, and thus the attribute of price is included. For example, for an undergraduate university student living on a study grant, €100 for software or an application is way too much, whereas for an industrial boss it is only pocket money.

4.5.16 Reliability

"The quality or state of being reliable." [107]. Reliability is a pragmatic UX attribute and means that the application is able to maintain its functionality in all situations and is able to recover or can be recovered from defects, infections and failures without losing information. Problematic situations may arise but they should be invisible, or nearly invisible, to the user (transparency). Software reliability is the major factor in software quality, since reliability problems can even make the whole system inoperative [102]. Naturally, people's opinions about reliability problems vary, but in general, it is not a good thing if the system is not reliable. In the modern technological world, reliability is tied to a period of time; a system is considered reliable enough if it remains functional over 100 days without a problem [53], or when considering environmental aspects, the device is considered reliable if it functions between -40 to +50 Celsius.

4.5.17 Stability

"The quality, stage or degree of being stable." [107]. While reliability means ability to maintain functionality in problematic situations, stability is about remaining stable over time without changes or modifications. In other words, problematic situations do not even arise. Stability is one of the most valued but hardest to achieve of the software quality attributes [104]. In order to inform users about possible instabilities in software, vendors or developers use alpha, beta or release candidate versions. Theoretically, software should never wear out, since it is only a set of instructions used to command an underlying device and there are no moving parts that might get broken. Thus, in theory, software is unbreakable. Unfortunately, this is not reality, and software products are very unlikely to remain stable over a long period of time [43]. Moreover, bones of software installed/downloaded by e.g., children and removed by fathers remain somewhere in the system, conflicting with other software, using resources and generally making PCs become slow and unresponsive. Instability is often a combination of hardware failures and software defects or infections. Initially, the number of instabilities caused by

software is higher, but it decreases over time as defects are identified and fixed. Over time, instabilities caused by worn hardware starts to increase and they can rarely be fixed with software changes.

4.5.18 Trust & Privacy

Trust: "*Assured reliance on the character, ability, strength, or truth of someone or something."* [107], Privacy (*"Freedom from unauthorized intrusion."*) [107]. Trust might mean questions like: Is this online shop trustworthy and is it safe to install this software. Privacy can be considered from many viewpoints. Sociologist view the concept as referring to social nuances, such as discussing private issues while riding on a crowded bus, while cryptologists see technical mechanisms such as data encryption and secured connections. This dissertation considers privacy from the technical viewpoint, which is close to trust, and therefore these two attributes were combined.

Privacy-affecting systems often succumb to at least some of these pitfalls, but if the system is able to avoid the above-mentioned difficulties, users are more willing to trust it and respect it with appropriate privacy practice. Many models for avoiding privacy and trust pitfalls have been developed, e.g., design patterns and utilization of mental models where a user mental model is aligned with designer information flow [98]. No matter how thoroughly trust and privacy issues are considered in software, the weakest link is the user. E.g. some people use unsecured WLANs without hesitation, because it is simpler than set up i.e. WPA2 (Wi-Fi Protected Access II) encryption. An alternative interpretation of their behavior might be that they are trusting. They feel the psychological (and other) costs of being distrustful outweigh the risks.

4.5.19 Usability

"*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.*" [69]. Usability can be defined by five quality components; learnability, efficiency, memorability, errors and satisfaction [123], which could all be seen also as independent parts of UX. There is a great difference in important usability aspects between users with different demographics and abilities. For example a study by Alakärppä et al. [3] considered a pain monitoring system used by both nursing personnel and patients. While the nurses had no difficulties in using the system, the authors of the study state that in order to be successfully implemented, the system must take into account the physical and psychological capacities of patients [3]. Despite usability and UX containing many of the same things, usability is only a part of UX and not interchangeable with UX. An example will make the difference clear:

> Someone has found the web page of a nice restaurant with a lake view. The web page includes an online reservation system that the person uses to book a nice corner table with a view of the lake. There is also an option to order and pay for meals beforehand, to avoid waiting in the restaurant and to get a 10 % discount. The person takes advantage of this possibility and finally prints a confirmation of the order. The usability of the website was excellent and everything worked flawlessly. When the person arrives at the restaurant, there is a 'full' sign on the front of the door. Knowing they have a reservation, the person enters the restaurant.

The doorman checks the reservation on the computer and is unable to find it, which leads to some argument and the arrival of the restaurant manager. The restaurant manager is very friendly and discovers that the newly-built reservation system has malfunctioned. However, she is unable to do anything about the situation since the system is built and maintained by a third party supplier. As the restaurant is full and the reservation cannot be verified and it is a weekend (the software supplier office is closed) the restaurant manager has no other option but to deny entry. The person is given a number to call on Monday to reclaim money paid. Regardless of how good the usability of the reservation page, the UX from the point of view of the user is probably very negative. Irrespective of who is at fault, the user is unlikely to consider going to the restaurant again

4.5.20 Usefulness

"The quality of having utility and especially practical worth or applicability." [107]. Usefulness determines whether the application, device, etc, can be used to achieve some desired goal or fulfill the purpose for which it was made. As in the case of functionality, what is useful for someone might be useless for someone else. Therefore this attribute is considered as both, pragmatic and hedonic. Despite being close to usability, according to, e.g. Nielsen [123] a product can be useful even if its usability is terrible. Usefulness is very difficult to evaluate beforehand [132]. For example, designers and UX professionals might consider a feature very useful, but actual users may think otherwise. In such a situation, who is to be believed? The dilemma behind this controversial issue has been illustrated by a study conducted by Root and Draper [132]. They asked participants to evaluate usefulness before and after use and discovered that there was no correlation between the before and after results.

4.5.21 User interface

"*The way a person interacts with a computer or electronic device*"[29]. Good UI is considered important for several reasons. If the UI is good, it will also be less expensive, since the amount of training, support, etc. is reduced [171]. Furthermore, if the UI is good, users (employees) like to use the device or application and their satisfaction and productivity are increased, which naturally leads to increased profits from the company viewpoint. Users, in general, are not interested in new coding techniques, object-oriented programming language, or flawless memory management [26]. The interesting thing for them is that the application meets their needs and works. According to authors' observations, for most users without technical experience, UI equals the system. After the first impression gathered from commercials, friends, package etc, UI is the next thing that a user sees when (s)he starts to use the actual system. Therefore UI either further enhances the first impression, or ruins it. In some cases UI can even be the first impression, i.e. in a business environment where IT-support performs all installations. First impressions should always be good ones, since it is difficult to transform a lousy first impression into a good experience [125]. This statement is generalizable in virtually everywhere, social contacts, web-sites, open houses, etc. On the other hand, a good first impression may easily be changed into disappointment, as underlying bad behavior cannot remain hidden for long. Building a

29 http://www.pcmag.com/encyclopedia/

beautiful UI for an application that is nonfunctional can be compared to putting an Armani suit on Attila the Hun [26].

4.6 UX attributes – results

Hypotheses were made for every UX attribute included in the surveys about the effect of demographics and ability to use computers. Age, gender and ability to use computers were all given the same 0-hypothesis (H0) and hypothesis (H1).

Table 3: UX attribute hypotheses

UX attribute	Hypotheses	
Accessibility Aesthetics Brand Consistency Coolness Device Ease of taking into use Emotions Environment Expectations Functionality Interaction Motivation Pleasure Price Reliability Stability Trust & Privacy Usability Usefulness User interface	Gender	H0:Gender and attribute are independent
		H1:Gender and attribute are dependent
	Age	H0:Age and attribute are independent
		H1:Age and attribute are dependent
	Ability to use computers	H0:Ability to use computers and attribute are independent
		H1:Ability to use computers and attribute are dependent

All numerical values and graphical presentations are based on the following transformation pattern.

- Very important = 1
- Important = 2
- Moderately important = 3
- Of little importance = 4
- Unimportant = 5

In other words, a lower value means that the UX attribute is more important and a higher value means the attribute is less important. Before presenting any analyses, it must be mentioned that UX professionals were asked to evaluate the importance of the UX attributes for themselves as users, and not to give their professional opinions about the importance of UX attributes for any particular target group. If they had been given a

target group based on which they should make their evaluations, the results might be closer to ones given by actual end users.

The responses were first handled without considering demographic variables or abilities; instead an average was calculated for 1420 end users (end user survey + upper secondary survey) and 12 UX professionals. Chart 2 presents this calculated average importance to UX professionals and end users. Results from the upper secondary survey have been included in the end users when applicable, since they are just another age group of possible end users. Original scale was 1-5 but 5 was dropped out from the chart since not a single attribute average was between 4 and 5.

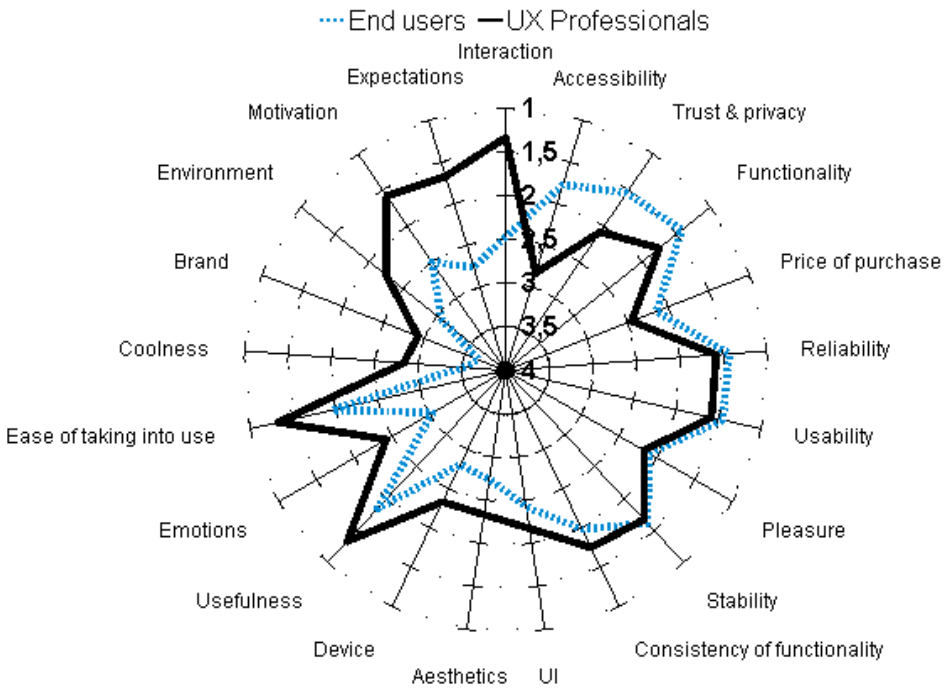


Chart 2: Average importance of all UX attributes by UX professionals (12 respondents) and end users (1420 respondents) (Closer to outer edge → more important)

If the most important attributes in the eyes of end users (fine dashed blue line; usability, reliability and functionality) are compared against the same attributes from UX professionals (black line) the differences between average answers are small. This indicates that UX professionals seem appreciate these items similarly to end users, when the group of end users is seen as one. On the other hand, big differences can also be found. For example, expectations, interaction and motivation are all highly appreciated by UX professionals, but end users in both groups seem to be less enthusiastic towards these attributes. Again it needs to be stated that UX professionals were asked to evaluate the importance of attributes for themselves, and not for a specific target group. Therefore, it cannot be said was the reason behind differences due to professionals vs.

end user or just because of personal demographics of professionals and end users. Also the sample size was small, 12 UX professionals. To be able to clarify further the important UX attributes for end users with different demographics and abilities, SPSS (Statistical Package for the Social Sciences) statistics[30] is taken into use. Methods, tests and analyses utilized under SPSS are based on personal communication with mathematical statistics teacher. First, the effect of age, gender, and ability to use computers is considered independently and secondly the combination of given properties is evaluated by using the *Wizard of UX* utility developed and prototyped for this dissertation. Chapter 5 describes more thoroughly the prototyping and evaluation of the utility.

4.6.1 Influence of gender

First, the influence of gender is considered between 814 end users males (continuous cyan line) and 601 end user females (dashed magenta line). In order to be able to compare end user averages against the averages from UX professionals also their averages were added to the charts. There are only three (3) UX professional male respondents (continuous blue line) and ease of taking into use attribute contains zero responses from males, so the results cannot be generalized. Average for nine (9) UX professional female respondents (continuous orange line) was also calculated. If the average of all included UX attributes is calculated for all four groups, the divergence is small. The average for both end user genders is 2.31 and for UX professional males 2.34 and females 1.92. Some individual attributes are valued identically by both end user groups regardless of gender, see right-hand side of Chart 3. The left-hand side of Chart 3 presents the most dissimilar UX attributes by end user males and females and compares those to the same values for UX professionals.

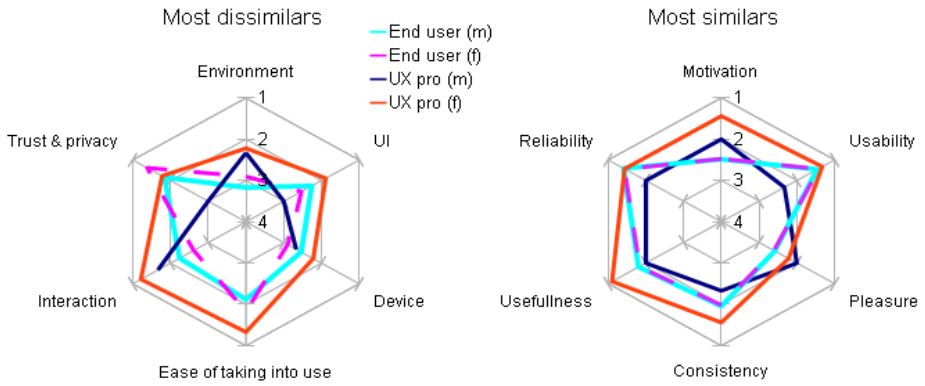


Chart 3: Most similar and dissimilar UX attributes by end user males and females

When the average of all end user male and female answers are calculated for each UX attribute, it seems that both genders appreciate motivation, usability, pleasure, consistency, usefulness and reliability similarly (cyan line and dashed magenta line

30 http://www-01.ibm.com/software/analytics/spss/

overlap). Some differences to UX professionals can be seen. When the most dissimilar UX attributes by end user females and males are compared, the differences are still small (0,2-0,4) when averages are considered. If, for example, the trust and privacy value for end user females is compared against the same value from UX professionals males the difference seems to be noticeable. The difference between end user females and UX professional females in interaction seems to be noticeable. Unfortunately statistical significance for these results cannot be demonstrated due to small amount of UX professional respondents.

To further clarify the differences between males and females it was decided to utilize the crosstabs analyses in SPSS. The test is used to demonstrate dependencies or in-dependencies between independent variable (gender) and dependent variable (surveyed UX attribute(s)). In addition the one-way ANOVA (Analysis of Variance) analysis was utilized to determine whether there are significant differences between males and females. Table 4 shows the calculated Pearson chi-square value from the crosstabs test and the associated risk of being wrong value in case the 0-hypothesis is rejected (Asymp. Sig column). Value of .05 was used in this column as divider between maintaining and rejecting the 0-hypothesis. The fourth and fifth column present the important results from the one-way ANOVA test. The fourth column shows the homogeneity of variance, in which the value must be greater than .05 in order to use ANOVA significance. In other cases the significance is taken from the Welch test of equality of means. In the fifth column, values under .05 mean that statistically significant difference between tested groups was found. Sixth column shows what to do with the 0-hypothesis, if the 0-hypothesis is rejected complete row is bolded and if Pearson and one-way ANOVA suggest different decisions, the result in ANOVA column is highlighted. The final column (result) shows if the attribute according to these results seems to be dependent or independent on gender.

Table 4: Pearson Chi-Square test; gender - UX attributes

UX attribute	Pearson Chi-Square	Asymp. Sig.	Homogeneity of variances Sig.	ANOVA Sig. / Welch Sig.	0-hypothesis	Result
Accessibility	**36.485**	**.000**	.666	**.000**	**Reject**	**Dependent**
Aesthetics	**25.191**	**.000**	.144	**.000**	**Reject**	**Dependent**
Brand	**31.223**	**.000**	.958	**.000**	**Reject**	**Dependent**
Consistency	2.377	.667	.755	.235	Maintain	Independent
Coolness	**12.326**	**.015**	.232	**.001**	**Reject**	**Dependent**
Device	3.458	.484	.138	.693	Maintain	Independent
Ease of taking into use	**56.706**	**.000**	.000	**.000**	**Reject**	**Dependent**
Emotions	**13.308**	**.010**	.161	**.005**	**Reject**	**Dependent**
Environment	**27.145**	**.000**	.305	**.000**	**Reject**	**Dependent**
Expectations	7.387	.117	.220	.048	Maintain	Independent
Functionality	3.687	.450	.801	.793	Maintain	Independent

Interaction	**38.970**	**.000**	.000	**.000**	**Reject**	**Dependent**
Motivation	**15.954**	**.003**	.023	**.000**	**Reject**	**Dependent**
Pleasure	3.578	.466	.566	.617	Maintain	Independent
Price	5.908	.206	.007	.216	Maintain	Independent
Reliability	4.376	.357	.824	.678	Maintain	Independent
Stability	2.698	.610	.922	.871	Maintain	Independent
Trust & Privacy	**39.284**	**.000**	.000	**.000**	**Reject**	**Dependent**
Usability	**15.908**	**.003**	.004	**.004**	**Reject**	**Dependent**
Usefulness	4.449	.349	.226	.379	Maintain	Independent
User interface	**24.245**	**.000**	.000	**.000**	**Reject**	**Dependent**

Results from both tests; Pearson and one-way ANOVA tell the same thing about rejecting or maintaining the 0-hypothesis. Only difference, highlighted in the table, can be seen in attribute expectations, but it was decided to maintain the 0-hypothesis thus the ANOVA significance is really close to .05 limit. When these findings are compared with the theory and research in social sciences and psychology, results are similar. For example, it has been shown that gender has an effect on product branding and aesthetic opinions [112], [113], [114]. These studies were concerned with how females and males appreciate different brands, e.g. Lancôme vs. Sony, or aesthetic designs. For the Lancôme vs. Sony question, not much second sight is needed to be able to tell that there is a clear favorite for both genders. Both brands are equally known by most people, but without experience of use they cannot be appreciated. The surveys in this study were about the importance of different UX attributes for overall UX when a certain application or device is used. Respondents were not given brands or different designs for evaluation. Instead, they had to present general opinions about the importance of the attributes without reference to specific brands. Yet, it was encouraging that the results from this study were similar to theory. For both UX attributes; aesthetics and brand, the 0-hypothesis was rejected with 0 % risk of being wrong. For attributes of consistency, device, expectations, functionality, pleasure, price, reliability, stability and usefulness gender according to the conducted Pearson chi-square was not found to be affecting demographic and the 0-hypothesis was maintained. In total of 12 UX attributes the gender was found to an affecting demographic.

Results from the conducted tests show that gender should be considered when software is designed. For example, if it is clear from the beginning of the development project that the main target group will be females, then some extra effort should be put into areas important for female users. Thus, reducing investment in interaction design and putting those resources into ensuring trust and privacy issues are correctly addressed might produce happier female users in the long run.

4.6.2 Influence of age

Again the crosstabs analysis in SPSS was utilized to measure the effect of age. To verity the results from the crosstabs test, the one-way ANOVA analysis was used. Further on

the Tukey and the Games-Howell post-hoc tests were used to find the age groups that have biggest differences according to this sampling.

The end user respondents were divided into six age groups according to divisions used by Statistics Finland (footnote 24 page 48). In the analyses, the following age groups were used; number of respondents in each group is in parenthesis after the group; 15-19 (78), 20-24 (564), 25-29 (424), 30-34 (173), 35-39 (61), 40-44 (52), and the rest as one group (68). Similar approach as in the gender case was utilized and the results are shown in a Table 5.

Table 5: Pearson Chi-Square test; age - UX attributes

UX attribute	Pearson Chi-Square	Asymp. Sig.	Homogeneity of variances Sig.	ANOVA Sig. / Welch Sig.	0-hypothesis	Result
Accessibility	25.601	.374	.231	.437	Maintain	Independent
Aesthetics	26.901	.309	.008	.523	Maintain	Independent
Brand	35.083	.067	.005	.086	Maintain	Independent
Consistency	36.307	.051	.604	.093	Maintain	Independent
Coolness	**36.860**	**.045**	**.965**	**.009**	**Reject**	**Dependent**
Device	**40.226**	**.020**	**.016**	**.037**	**Reject**	**Dependent**
Ease of taking into use	**43.224**	**.009**	**.259**	**.013**	**Reject**	**Dependent**
Emotions	24.475	.453	.648	.038	Maintain	Independent
Environment	29.059	.218	.020	.098	Maintain	Independent
Expectations	**51.876**	**.001**	**.023**	**.000**	**Reject**	**Dependent**
Functionality	21.710	.597	.050	.164	Maintain	Independent
Interaction	**39.172**	**.026**	**.093**	**.000**	**Reject**	**Dependent**
Motivation	27.916	.264	.048	.284	Maintain	Independent
Pleasure	**37.413**	**.040**	**.250**	**.002**	**Reject**	**Dependent**
Price	**46.069**	**.004**	**.599**	**.000**	**Reject**	**Dependent**
Reliability	23.694	.479	.000	.046	Maintain	Independent
Stability	16.785	.858	.001	.196	Maintain	Independent
Trust & Privacy	21.781	.592	.380	.681	Maintain	Independent
Usability	25.785	.364	.096	.591	Maintain	Independent
Usefulness	16.099	.884	.552	.927	Maintain	Independent
User interface	**44.698**	**.006**	**.000**	**.007**	**Reject**	**Dependent**

The Table 5, bit surprisingly, indicates that the effect of age on UX attributes is lower than in a case of gender. 0-hypothesis was rejected for only eight UX attributes and 0.00 % risk of being wrong was not found for any attribute. On the other hand, the table shows that UX attributes that are directly related to using some software like usefulness,

stability and functionality are not dependable of age, a finding that follows common sense. For attributes emotions and reliability, the Pearson correlation and the one-way ANOVA suggest different decisions, highlighted in table, for 0-hypothesis, but the significance of latter was relatively close to .05 limit, thus the result from the Pearson is accepted. For the remaining 19 attributes both test suggest the same decision.

It has been claimed that perspectives towards life changes with age [156], and that denial of age is fairly common at some stage of the human life cycle. Sykorova states that age is a relevant factor in various contexts and an important aspect of an individual's identity [156]. The results of the analyses performed for this thesis further enhance her statements since e.g., appreciation on attributes of coolness, expectations and pleasure were dependable of age. Table 6 below further explains which groups differed from each other in cases where 0-hypothesis was rejected. For attributes in which the homogeneity of variances was greater than .05 the Tukey post-hoc test was used and for the rest, Games-Howell post-hoc test was used.

Table 6: Most significant differences between age groups

UX attribute	Used test	Most differed groups	Sig.
Coolness	Tukey	20-24 vs. 30-34 20-24 vs. over 44	.086 .072
Device	Games-Howell	15-19 vs. over 44	.047
Ease of taking into use	Tukey	20-24 vs. 40-44 20-24 vs. over 44	.045 .052
Expectations	Games-Howell	20-24 vs. 35-39 20-24 vs. 40-44 25-29 vs. 35-39 25-29 vs. 40-44	.030 .013 .006 .000
Interaction	Tukey	20-24 vs. 25-29	.000
Pleasure	Tukey	20-24 vs. 30-34 20-24 vs. 35-39	.005 .021
Price	Tukey	15-19 vs. 30-34 15-19 vs. 35-39 15-19 vs. over 44 20-24 vs. 30-34 20-24 vs. 35-39 25-29 vs. 30-34 25-29 vs. 35-39	.022 .016 .074 .001 .016 .023 .069
User interface	Games-Howell	20-24 vs. 25-29 20-24 vs. 30-34	.002 .085

As an example, calculated differences for the attribute price were calculated. Table 7 shows the results. Number of respondents is shown in parenthesis after the age group.

Table 7: Differences for attribute price when age is considered

Age group	15-19 (78)	20-24 (564)	25-29 (424)	30-34 (173)	35-39 (61)	Over 44 (68)
Mean	1.78	2.18	2.24	2.55	2.66	2.52
Std. Dev.	.831	1.07	1.05	1.06	1.05	1.1
Std. Error of mean	.181	.046	.051	.081	.134	.139
Variance	.690	1.15	1.10	1.12	1.10	1.21

When individual answers were combined into averages for age groups, interesting findings were made. For example, appreciation of brand in 15-19 year old is 3.5, drops to 4.0 (in the 30-34 age range), starts to rise again, and is 3.7 in the group of respondents over 44 years old. It is only possible to speculate about the reasons for this behavior, but according to [62] brand and model are important for staying in a 'circle of friends'. The news story considered respondents from 13 to 17 so it included comprehensive school students also. Nevertheless it might be the cause for higher appreciation towards brand at lower ages. On the other hand, the higher appreciation in older ages may be explained by the fact that older respondents have used the same brand or model for a longer period and have already bonded to it. In spite statistical analyses saw age and brand as independent, results were close to limits and if the 0-hypothesis would have been rejected the risk of being wrong would have only been 6,7 %.

To be able to see if age has a similar effect with UX professionals, their values are calculated. The attributes of pleasure and UI were the only ones in which the 0-hypothesis would have been rejected (Asymp. Sig ,046). Still, the values for UX professionals can only be considered suggestive since the data only contained 12 responses.

The results suggest that the age of primary end users should be considered during the software development process. If development efforts are directed in the right way from the beginning, considerable time and money might be saved by avoiding unnecessary work. The data do not include children, teens or elderly people, so their attitude to the UX attributes studied remains unknown. It could be speculated that the same trend as in the case of attribute price would continue and probably even strengthen. It would be interesting to test this hypothesis in future research.

4.6.3 Influence of ability to use computers

The used scale of ability to use computers was presented in section 4.3 . In the analyses, the very poor alternative (1) was omitted since the category only contains three responses. Amount of respondents on other levels are: 2=92, 3=634, 4=424, 5=146 and 6=61. The ability of UX professionals to use computers was not considered since the variation in their ability was very small; seven respondents at level 4 and five respondents at level 3. Same approach as in the gender and age case was used to test if

the 0-hypothesis for end users should be maintained or rejected. Table 8 shows the results.

Table 8: Pearson Chi-Square test; ability to use computers - UX attributes

UX attribute	Pearson Chi-Square	Asymp. Sig.	Homogeneity of variances Sig.	ANOVA Sig. / Welch Sig.	0-hypothesis	Result
Accessibility	58.707	.000	.002	.217	Maintain	Independent
Aesthetics	23.058	.112	.195	.053	Maintain	Independent
Brand	18.792	.280	.763	.397	Maintain	Independent
Consistency	14.891	.533	.680	.095	Maintain	Independent
Coolness	**36.521**	**.002**	**.430**	**.000**	**Reject**	**Dependent**
Device	**36.445**	**.003**	**.002**	**.118**	**Reject**	**Dependent**
Ease of taking into use	**71.245**	**.000**	**.000**	**.000**	**Reject**	**Dependent**
Emotions	**29.307**	**.022**	**.629**	**.019**	**Reject**	**Dependent**
Environment	**33.992**	**.005**	**.002**	**.001**	**Reject**	**Dependent**
Expectations	12.049	.741	.416	.514	Maintain	Independent
Functionality	13.820	.612	.043	.107	Maintain	Independent
Interaction	**110.436**	**.000**	**.009**	**.000**	**Reject**	**Dependent**
Motivation	21.373	.165	.984	.027	Maintain	Independent
Pleasure	14.468	.564	.887	.920	Maintain	Independent
Price	**65.837**	**.000**	**.000**	**.000**	**Reject**	**Dependent**
Reliability	23.500	.101	.007	.050	Maintain	Independent
Stability	**29.007**	**.024**	**.000**	**.054**	**Reject**	**Dependent**
Trust & Privacy	**42.495**	**.000**	**.000**	**.006**	**Reject**	**Dependent**
Usability	14.143	.588	.246	.834	Maintain	Independent
Usefulness	23.399	.063	.062	.621	Maintain	Independent
User interface	**46.022**	**.000**	**.000**	**.000**	**Reject**	**Dependent**

Four contradictions, highlighted in table, between the Pearson correlation and the one-way ANOVA test were found. It was decided to accept the results from Persons if its risk of being wrong was under .05 and the result from the one-way ANOVA test was under .20. If the Pearson value was over .10 and ANOVA under .05, the Pearson was accepted. To more clarify the differences between exact groups, same tests were run as in the case of age; Tukey post-hoc and Games-Howell post-hoc. The results are shown in a Table 9. Averages of bolded attributes are presented after the table in Chart 4.

Table 9: The most significant differences between ability groups

UX attribute	Used test	Most differed groups	Sig.
Coolness	Tukey	2 vs. 3 2 vs. 4 2 vs. 5	.012 .000 .005
Device	Games-Howell	2 vs. 4	.079
Ease of taking into use	Games-Howell	2 vs. 3 2 vs. 4 2 vs. 5 2 vs. 6 3 vs. 4 3 vs. 5 3 vs. 6	.031 .000 .000 .002 .000 .000 .074
Emotions	Tukey	3 vs. 5	.056
Environment	Games-Howell	2 vs. 4 2 vs. 5 2 vs. 6 3 vs. 5	.027 .003 .060 .020
Interaction	Games-Howell	2 vs. 3 2 vs. 4 2 vs. 5 2 vs. 6 3 vs. 4 3 vs. 5 3 vs. 6 4 vs. 5.	.015 .000 .000 .000 .000 .000 .000 .058
Price	Games-Howell	2 vs. 5 3 vs. 5 3 vs. 6 4 vs. 5	.011 .000 .034 .001
Stability	Games-Howell	No differences were found	
Trust & Privacy	Games-Howell	2 vs. 4 2 vs. 5	.028 .003
User interface	Games-Howell	2 vs. 4 3 vs. 4	.019 .000

The information collected reveal that price, environment and ease of taking into use attributes loss some of the importance when ability to use computers is higher while for example, interaction becomes more important when ability to use computers is higher. This can be explained by raised professionalism, ability to 'override' the context aspects and ability to maintain the focus on the application or task at hand. Users with professional level skills are willing to pay for a good application and manually configuring some settings before the first use will not probably form a problem for them. At the same time, for the user with a lower ability to use computers, these configurations would be probably unachievable.

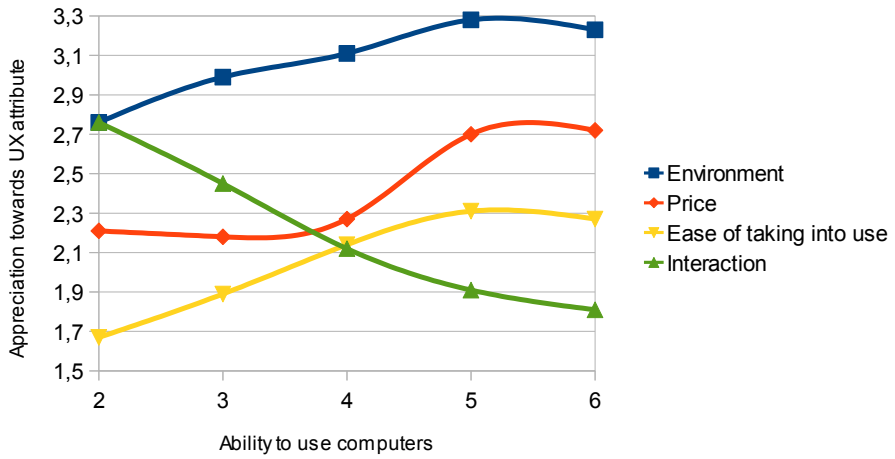


Chart 4: Effect of ability to use computers

In addition to attributes presented in the chart, it needs to be mentioned that 17 of all 21 UX attributes produced an absolute correlation value over 0.5, which according to Laerd statistics [94] can be viewed as good strength of association. Smallest correlation was found for expectations; the value was -0.39, which according to [94] is medium strength of association. One noteworthy finding is that when ability to use computer increases, respect for functionality and consistency of functionality is valued more. This may be explained by the fact that when ability to use computers is high, then the full potential of an application is commonly put into use and features such as keyboard shortcuts and professional functionality are utilized. For example, if a Windows application contains a possibility to cut and paste or copy and paste information, advanced users will assume that ctrl + X, ctrl + C and ctrl + V can be utilized. For many users, having to use the Edit menu for this functionality is very awkward. The same holds for other keyboard shortcuts and application features; once users with high ability to use computers have learnt to use a function, they naturally expect that it can be utilized elsewhere as well. Usability and stability issues seem to be equally valued, regardless of the level of ability to use computers. The value ascribed to usability varies between 1.42-1.48, with an average deviation of only 0.01, and stability varies between 1.25-1.43, with an average deviation of 0.05. Despite being numerically close to each other, it might be the case that usability means different things for users in ability levels 2 and 6

These findings indicate that ability to use computers has an impact on appreciation of many UX attributes. According to statistical analyses appreciation toward ten attributes; coolness, device, ease of taking into use, emotions, environment, interaction, price, stability, trust & privacy and user interface, is dependable of ability to use computers. Therefore, all software development projects should take into account the target groups' ability to use computers when UX of end users is considered.

4.6.4 Combination of demographics and abilities

Of the three demographic variables included in the study, gender was found to be the most meaningful; twelve 0-hypotheses were rejected. When age was used as divider, eight 0-hypotheses were rejected and on ability to use computers case, ten 0-hypotheses were rejected. When all the results are considered together, it seems that four UX attributes are dependent on all surveyed demographics and abilities when the statistical significance is considered, these attributes are:

- coolness
- ease of taking into use
- interaction
- user interface.

There were also four UX attributes that, according to statistical analyses, were not related to any of the used demographics or abilities. These attributes are:

- consistency
- functionality
- reliability
- usefulness

Not surprisingly all 'not related' attributes can be connected to using some application and according to results it can be stated that these attributes are valued equally in spite of the demographics or ability to use computers.

When all 21 UX attributes were considered as one user experience (calculated average), virtually no difference was seen. However, when individual attributes were considered, differences were identified. This indicates that to utilize the full potential of demographics and abilities of end users, it is required to use greater details. SPSS is a superior tool for analysing information like this, but it is unlikely that a software developer would use this or similar software. Furthermore, small development companies might not possess resource to purchase this software. Therefore, rests of the analyses are conducted with software developer oriented way; fastly and cheaply by utilising the *Wizard of UX*.

In Table 10 each surveyed UX attribute is presented for two different groups; the one that showed greatest appreciation of the particular UX attribute and the one that showed least appreciation of the same UX attribute. The results for all groups are presented in columns which follow the format; **'gender / age / ability to use computer (average importance)'**. If gender, age or ability to use computers were not found to be an affecting factor, then the cell is left empty. If two different combinations happened to

produce exactly the same value, then the one with more respondents was chosen for the table. The DN column stands for 'I do not understand / I do not know' answers and is presented at the right-hand side of the table. The table is sorted according to the difference between the greatest and least appreciation (Diff. Column). Dependent UX attributes from this subsection are highlighted with cyan colour and independent with orange colour.

Table 10: Most and least appreciation of UX attributes by end users according to the *Wizard of UX*.

UX attribute	Most appreciation Gender/age/ca (ave)	Least appreciation Gender / age / ca (ave)	Diff.	DN
Pleasure	M/<24/ (2,54)	F/>35/ (2,87)	0,33	1,9%
Stability	F/ /4 (1,18)	F/<24/1-2 (1,65)	0,47	0,3%
Accessibility	F/>35/1-3 (1,66)	>44/1-2 (2,33)	0,54	3,4%
Usability	F/35-44/1-3 (1,19)	M/>44/1-3 (1,79)	0,6	0,9%
Reliability	M/<20/ (1,20)	F/20-24/2 (1,94)	0,74	1,3%
Expectations	F/35-44/4-6 (2,08)	M/25-29/3-4 (2,88)	0,8	1,3%
Environment	F/30-34/1-2 (2,69)	F/ /6 (3,5)	0,81	1,6%
Brand	F/25-29/4 (3,32)	F/35-44/4-6 (4,17)	0,85	0,9%
Usefulness	F/35-44/4-6 (1,50)	M/>35/3 (2,36)	0,86	1,7%
Motivation	F/<24/5-6 (2,14)	M/>35/1-2 (3,00)	0,86	1,8%
Emotions	F/<20/1-3 (2,69)	M/30-34/5 (3,55)	0,86	1,8%
Trust & Privacy	F/ /6 (1,00)	M/<24/6 (1,91)	0,91	0,4%
Aesthetics	F/25-29/4-5 (2,92)	M/>44/1-3 (4.00)	1,08	17,1%
Consistency	F/30-44/5-6 (1,50)	F /<24/1-2 (2,69)	1,19	7,3%
Ease of taking into use	F/35-44/1-3 (1,59)	M/20-24/6 (2,91)	1,32	0,7%
Functionality	F/35-44/4-6 (1,17)	M/>30-44/1-2 (2,5)	1,33	0,70%
Coolness	M/<20 (3,00)	F/>44/5-6 (4,60)	1,6	1,30%
User interface	M/25-29/4-5 (1,84)	F/<24/1-2 (3,85)	2,01	8,4%
Interaction	F/25-29/5-6 (1,20)	F/<24/1-2 (3,30)	2,1	3,2%
Price	M/<20/3-4 (1,56)	F/>35/5 (3,80)	2,24	1,8%
Device	M/<20/3-4 (2,11)	F/>35/5 (4,40)	2,29	2,9%

Statistically significant effect of age, gender or ability to use computers were not found for all UX attributes, but still only two attributes with a difference value of less than 0.5 can be seen in the table. For ten attributes, the difference between the most and least appreciating group is between 0.5 and 1.0. Six attributes remain under 2.0 and for the rest the difference is over 2.0. The biggest difference can be spotted in device and price. In the case of interaction, however, it is rather surprising that the groups with the greatest and least appreciation of the attribute are so close to each other; both groups are young females, the difference is formed by the ability to use computers. For the user

interface attribute, the group with the least appreciation of this attribute is the same group as for interaction, but the group which appreciates this attribute the most is Males/25-29/4-5. It might seem odd that the attributes price and device are not highlighted in the table. This is due to fact that SPSS did not find dependency between gender and price or device. This can be explained with the utilized analyses in SPSS, which considered gender, age and ability to use computers independently from each other.

The data in Table 10 further demonstrates that demographic variables of end users and the ability to use computers have an effect on appreciation of different UX attributes. The table does not consider the amount of respondents in presented groups, but the *Wizard of UX* is implemented in a way that it will not show the results if there are under five respondents.

Finally it needs to be clarified that use of a basic spreadsheet program like Excel or Calc to analyse this information, would have been extremely laborious. The *Wizard of UX* utility was prototyped and utilized for this purpose.

4.7 Discussion

On the basis of the results from this chapter, it can be concluded that demographic variables and ability to use computers have an effect to appreciation towards many UX attributes. Additionally, according to results collected from this study it seems that from the surveyed demographics and abilities, gender is the most meaningful one and a bit surprisingly age the least meaningful. From the surveyed 21 UX attributes four (coolness, ease of taking into use, interaction and user interface) were dependent on gender, age and ability to use computers also four (consistency, functionality, reliability and usefulness) seemed to be independent.

In addition, knowledge about target groups and their demographics and abilities were presented. The analyses provided by this chapter meets objective O1 (*Raising the UX knowledge among software developers.).* Lastly, this chapter provides the data for the *Wizard of UX* database which in combination with the *Wizard of UX* meets objective O2 (*Considering the attributes of UX by utilizing statistical user profiles in the early phases of software development.).*

5 Design and prototype of the *Wizard of UX*

The information gained from the surveys presented is of little value unless there is a simple way to utilize it. One alternative is to use spreadsheet program, but the greater the number of attributes, the more laborious the analyses becomes. This chapter presents the designed and prototyped *Wizard of UX*, which offers a simple way to benefit from the UX knowledge gathered and offers an effective way to improve knowledge of UX among software developers.

The *Wizard of UX* is a user experience utility, which provides an insight into what user experience attributes end user target groups with different demographics appreciate and compares the results against qualitative knowledge from UX professionals. The utility is constructed based on data from the publication presented in the Interact 2009 doctoral consortium [84]. The main beneficiaries of the *Wizard of UX* are software developers (programmers) and small development companies in which the programmers are responsible for nearly all aspects of development. If Cooper is to be believed, most programmers have little understanding of design for UX [26]. In smaller software development companies the main problem with UX is a lack of resources. Naturally, programmers with good knowledge and know-how about UX can also benefit from utilizing the *Wizard of UX*, since it offers concrete support for their previous experiences and theoretical knowledge. Even for an experienced UX professional, the *Wizard of UX* might offer new ways to assess UX from the perspective of end users, a task which is seen as difficult by many UX professionals [167].

Insights provided by the *Wizard of UX* are based on past knowledge from respondents with similar demographics as the current target group. In addition, general information about the UX attributes and their utilization is provided for people, especially programmers, without knowledge of UX. The *Wizard of UX* is a combination of two different parts; the *Wizard of UX* utility, and the user experience database. The application part is visible to users and the database is in the background, hidden from the users.

The *Wizard of UX* is implemented with regard to objective O3, *(To prototype and test a utility that assists in meeting O1 and O2.)*, and thus meets objectives O1 (*Raising the UX knowledge among software developers.)* and O2 *(Considering the attributes of UX by utilizing statistical user profiles in the early phases of software development.)*

As the developer of this utility (the author) was already familiar with Microsoft products, the decision was made to utilize Microsoft products in this part of the study. Table 11 presents the pros. and cons of decisions made before commencement of prototyping.

Table 11: Prototyping decisions with cons and pros

Decision	Pros	Cons	Requirements for the *Wizard of UX*
Utilize Microsoft products	Windows is the world most used OS, compatibility	Many programmers prefer Linux	Windows
Use C#	Simple, effective		
Use .NET framework 4.0 (later on a support for 3.x was added)	Offers new features	Requires manual installation if Windows < Vista	.NET framework 4.0 must be installed
Use database	Data is safe from the users	Firewall may cause problems	Requires network connection
Use Expression blend	More possibilities for the UI design		Might require blend related libraries
Individual application	Not tied to any particular IDE	Must be launched separately	

Designing and prototyping followed a 'beaver building its dam' procedure described by Cooper [26], and thus some decisions were made according to what was easiest to transform into code and changes made to the design after it was noticed that certain elements were too difficult to implement. Before going into further details, it needs to be emphasized that the *Wizard of UX* is currently a prototype used as a proof of concept application. The current version of the *Wizard of UX* utility can be downloaded, behind the link[31].

5.1 Design

A key consideration during the design was that the end users of the *Wizard of UX* are software developers who already have their hands full of work and do not have much time to learn new things. Therefore, the UI design was kept as simple as possible and all unnecessary features were omitted. Priority was given to the fact that the *Wizard of UX* was to work as a proof of concept application for this dissertation, which led to the following requirements:

- Ability to check the importance of UX attributes for users with different demographics and abilities.
- Ability to check the meaning of UX attributes.

31 http://tinyurl.com/24x3kpl Note! The Wizard of UX database server is requested to be operational until 31.12.2011. After this date, feel free to contact the author if you are interested in testing the *Wizard of UX*

5.1.1 UX of the *Wizard of UX*

Regardless of the field, success is always dependent on end users. For example, Alben has stated that professionals, even experienced ones, cannot substitute for users' personal involvement [4]. Therefore, the end user point of view should always be considered. In this case, the target group of the *Wizard of UX* is software developers, who are also the end users of the utility. On the other hand, software developers use *Wizard of UX* to check what end users of their projects appreciate. When software developers use this utility, their end users are not personally involved, but the demographics, abilities and knowledge about the importance of UX attributes of their end users are revealed. Therefore, the end users are implicitly involved, albeit at one step removed, and their knowledge will help software developers designing for UX. Software development always aims at satisfying some need and in the case of the *Wizard of UX* this need was primarily to act as a proof of concept application and secondarily to support the work of software developers. Ironically due to very tight deadline, in this dissertation, the UX of the *Wizard of UX* becomes a secondary target in those parts that do not serve the primary purpose.

The *Wizard of UX* does not participate in on-going discussions about the precise meaning and definition of UX by offering definitions since it is felt that users would probably use definitions and interpretations that fit their own needs. The same applies to consideration of theories dividing UX into multiple different methods. The *Wizard of UX* does not take into account if something is tied to a specific moment or to a longer period of time, nor does it divide attributes into hedonic or pragmatic categories. It just presents the attributes and their importance to a target group without adding unnecessary complications.

Making profit is the main intention of virtually every company. Although UX can be considered a sales promoter and a way to gain competitive advantage, the 'Is it worth it?' question should nevertheless be raised at all decision-making steps. Even if a company feels UX 'is worth it' and appreciates its importance, the company might not possess enough resources or knowledge to take advantage of these insights. To overcome and facilitate this issue, the *Wizard of UX* is free and simple.

5.2 *Wizard of UX* prototype

Many fancily-named UX evaluation methods, processes, etc. exist whose intention is to consider HCI and UX during the software development processes. Regardless of the method used or what it is called, the main purpose should be to design for good user experience [166]. Unfortunately, UX validations are commonly conducted within the group that is responsible for the designing for UX. This naturally decreases the trustworthiness of results. In addition, most of the evaluation methods will not work until something concrete can be used. This UX free area in the early phases of development discussed in Section 3.8. (see Figure 9) is the target of prototype. The intention of the *Wizard of UX* is to partly fill this target area by offering knowledge about important UX attributes on the basis of the demographics of future target groups. The knowledge behind the *Wizard of UX* is based on things that have already happened. Therefore, this utility does not ask end users to evaluate something; in fact, the actual

end users are not required in order to gain benefit from the utility. The only required things are end user demographics and their estimated ability to use computers. Based on this information, the *Wizard of UX* estimates their appreciation of various UX attributes utilizing pre-collected UX knowledge from similar users.

As mentioned earlier, Lord Kelvin highlighted the importance of numerical presentation [92], and this is the function for which the *Wizard of UX* is designed. The main functionality of the *Wizard of UX* is a possibility to present attitudes toward UX attributes in numerical form or with an illustrative graph.

Design mostly took place by creating a conceptual sketch of UI and transferring it into a real UI. Final UIs for the *Wizard of UX* were built with Expression Blend 4[32], which offers a wide variety of tools for this purpose. Functionality inside the UI was created with Visual Studio 2010[33] by utilizing the C# programming language. The *Wizard of UX* database was built with Excel and an SQL (Structured Query Language) server 2008[34]. The main reason for utilizing these particular tools was the fact that Webropol polling system was used for conducting surveys. Webropol is able to export results in Microsoft Excel format, and SQL servers are able to transform Excel sheets into SQL databases.

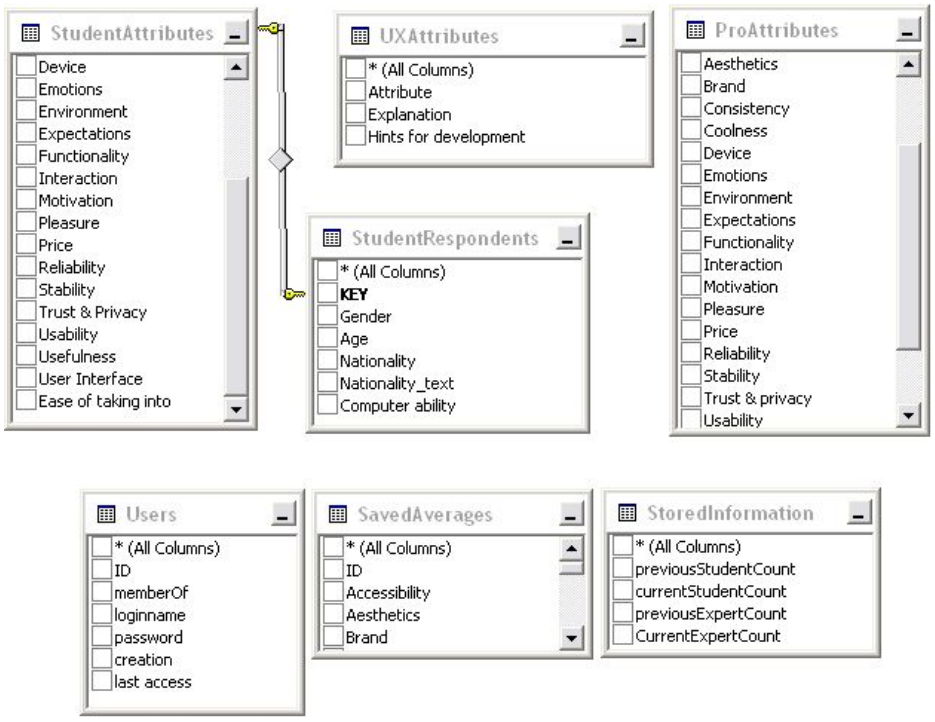
5.2.1 UX database

A new database was created inside the SQL server, and all rational and coherent data was inputted. The end user table was divided into two different parts; demographics and UX attributes. By using this division, the server does not have to go through one big table when the query arrives, instead it can go through a number of smaller demographics tables to pick correct lines and just retrieve the linked values from the attributes table. Currently, when the tables are small, the differences between these two methods remain somewhere between 10-100 milliseconds. In practice, the difference is unnoticeable, and the network connection causes more delays. If the tables grow, then the differences in query times would naturally also grow. Linking between the tables was carried out by adding the same bigint type key value for both members of the couple. This table division was not done for the UX professional responses since there are only 12 responses inside the table.

The UX definitions part was omitted from the database since the UX attributes offer a lot more knowledge about the interests of the target groups than a plain definition. In addition it was assumed that a software developer utilizing this tool does not require knowledge about how different UX definitions are appreciated. Instead, a data table with explanations of the UX attributes and hints about how this particular attribute could be taken into account during the development was added. This table is linked with the *Wizard of UX* in order to raise awareness and knowledge about UX among software developers. The information presented in this added table is based mainly on Section 4.5 of this dissertation. For possible future use, a table for user permissions was also created. Currently, as the database is used only for testing purposes with the *Wizard of*

32 http://www.microsoft.com/expression/products/Blend_Overview.aspx
33 http://www.microsoft.com/visualstudio/en-us/products/2010-editions
34 http://www.microsoft.com/sqlserver/en/us/default.aspx

UX, it relies on two pre-created test accounts and does not contain a possibility to add other users without manually altering SQL server settings. All of the built-in accounts of the SQL server were disabled and test accounts were only given restricted access to this particular database. This means that the test users are able to connect to a database and perform select queries, but, for example inserting and executing are forbidden, since those actions are not needed in order to use the *Wizard of UX*. In addition, a table for stored averages was created for future use. Currently, the largest data table is only about 1300 lines, so performance is rapid enough, even if it goes through the whole table with every query (about 140ms per query). For a table with millions of lines it would be extremely beneficial to calculate and store averages when a query arrives and to check first if pre-calculated values for that particular query exist. A table for storing the number of current respondents was also created to provide an easy way to check if the number of respondents had changed since the last average calculation. Figure 11 presents the structure of the current *Wizard of UX* database. The upper part shows the active tables and the bottom part shows the 'just in case' tables which are currently not used.



5.2.2 Using the *Wizard of UX*

Figure 12 presents the high level structure and operational chain of the *Wizard of UX*. When the *WizardofUX*.exe* file is clicked, normal Windows application launch

procedure begins. Somewhere during this process .NET framework files are loaded. Since this tool is a prototype, it was decided not to spend any time making framework checks during the launch. Therefore, the *Wizard of UX* will not run or might even crash if it cannot find an appropriate .NET framework.

When the required .NET framework version is found, the *Wizard of UX* launches. Then the *MainWindow* containing the C# application initializes and creates a connection to the database. If the database connection fails, then the Problem page is shown to the user with an explanation. In this version, the only way out from the Problem page is to close the *Wizard of UX*. The *MainWindow* contains menus, tabs, background images and base colours that are visible in all other pages except the Problem page. The layout of the *MainWindow* is based on a full sized grid, which contains different UI elements, like menus, and borders and a frame which shows the selected sub-page. If the database connection succeeds, then the main process is accessed and the default view is loaded inside the *MainWindow*.

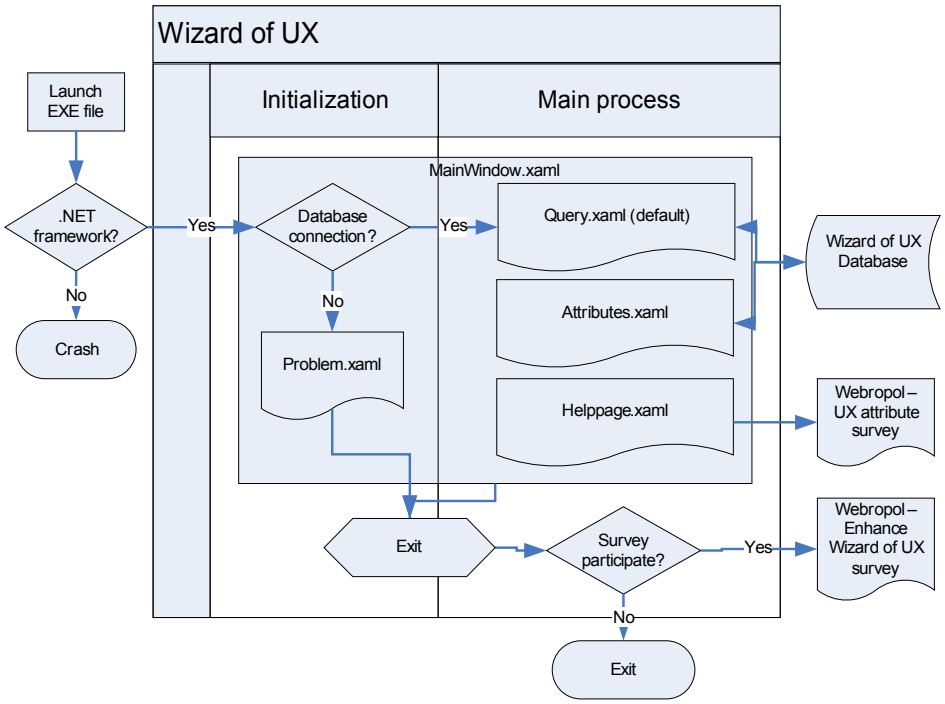


Figure 12: High level structure of the *Wizard of UX*

Query view

This is the default view of the *Wizard of UX* and is divided into two sides. On the left-hand side of the Query view, available end user demographics for the query are presented. The actual SQL query is formed automatically based on the selections when the 'Perform Query' button is pressed. When the query is performed, there are two

alternatives for the right-hand side of the query view. If there are less than five responses with the given parameters inside the database, a notification is shown that the user should redefine the query. If the number of responses is greater than five, then the results are listed. The top and bottom five results are shown with a graphical presentation for the chosen end user group and for UX professionals. Currently, the *Wizard of UX* will not use demographics for selecting UX professionals and the average of all 12 is shown. If a mouse cursor is hovered over the bar, a numerical value is shown. Figure13 presents a screen capture of this default view when the mouse is hovered over the Brand bar of end users. A single button click allows this to be changed to view all UX attributes and also allows the presentation to be changed into numeric format. This view responds to the objectives O2 and O3 by offering starting points for development by showing the important and unimportant UX attributes in the eyes of end users.

All queries are formed with SQL. The use of LINQ (Language Integrated Query) was not considered since the author was already familiar with basic SQL and the queries are only used for the SQL database. In the interests of simplicity, the whole SQL query is performed as a black box and only the result is shown to the user. This means that the actual SQL query will not be visible to the user at any point. The possibility to alter the SQL query manually is disabled in this version.

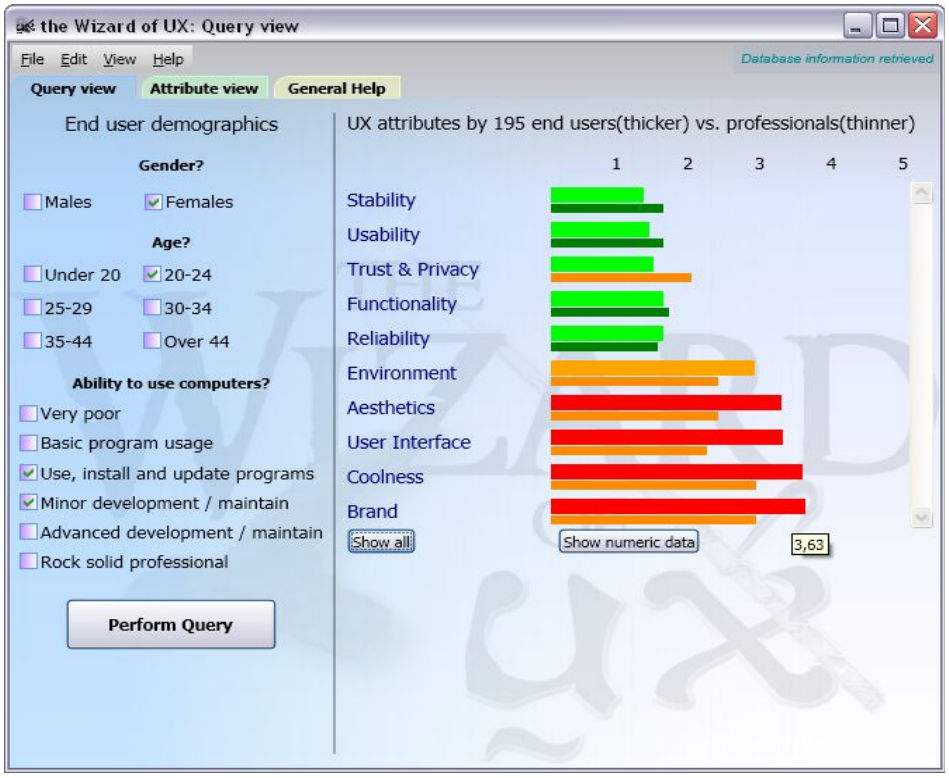


Figure 13: the *Wizard of UX* - query performed

Attribute view

The attribute view is divided into upper and lower parts. Inside the upper part, all UX attributes that are available in the *Wizard of UX* database are dynamically loaded when the view is accessed. UX attributes are presented inside boxes that behave like buttons. When the mouse is hovered over the box, it is highlighted with a greenish colour and explanations and hints, which are mainly based on Section 4.5, for the development and information about average appreciation by end users and UX professionals are shown in the lower part of the screen. In addition, a tip about clicking the box either with right or left mouse button is given. If the right mouse button is clicked over the box, a Google search about the attribute is launched with the results given inside a system default browser. When the mouse no longer hovers over the box, the information disappears and the colour is removed. It is also possible to lock the selection while hovering over the box by clicking with the left mouse button. This action colours the box with a bluish colour and makes it possible to copy UX information from the text fields in the lower part of the view. Figure 14 presents the attribute Aesthetics as locked. The lock can be released by clicking the box again. It is also possible to change the lock to another UX attribute by clicking its box. Changes in the locked UX attribute will naturally change the information in the lower part of the view as well. This view responds to objective O1 of offering knowledge about UX attributes to the user.

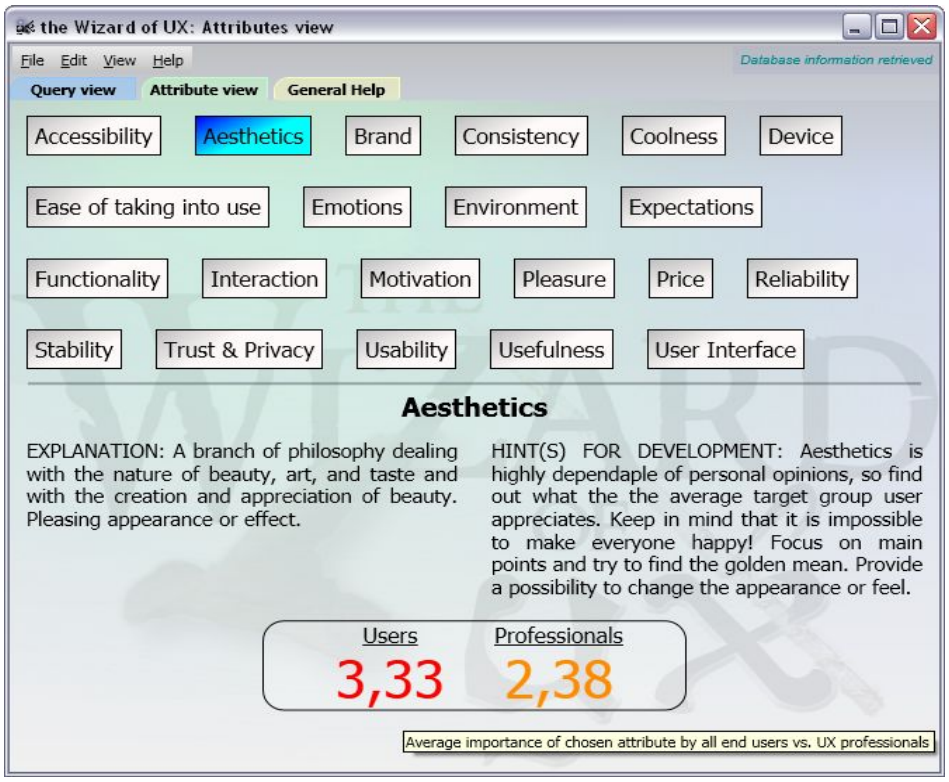


Figure 14: the *Wizard of UX* - aesthetics locked

Help view

The final view inside the main process is a general help view which contains information about the *Wizard of UX*, as well as instructions for moving between the views. This view also contains a link to a Webropol survey that can be used to produce more data for the UX database. Authors contact information is also presented under this view.

When a request about exiting the *Wizard of UX* is raised by utilizing any of the three possible methods (via file menu, by pressing X in the upper right corner or by pressing Alt+F4) a message box with a request to evaluate the *Wizard of UX* is shown. If a 'No' response is given, the application will exit. If 'Yes' is pressed, the application exits and a Webropol survey is launched.

All network connections from the *Wizard of UX* to the database are unsecured so it would be extremely simply to, e.g. capture a username and password. But as stated earlier, the test accounts are limited to connecting and performing select statements, so no harm can be done and no confidential or private information is transferred. The links to the Webropol surveys are also created without secure connection, but surveys are public anyway.

5.3 Testing the functional prototype

Testing happened in three phases, first the functionality of the *Wizard of UX* was tested by people familiar with software development. This test was conducted simultaneously by three persons; the author and two co-workers. Since this is a prototype and it is not expected to happen any time soon that the *Wizard of UX* will have simultaneous users, it was decided that testing with three people was enough see that the *Wizard of UX* is functional enough. During the test no infections or failures were detected or identified, after which it was proceeded to test the *Wizard of UX* in real use under programming oriented course and with professional UX designer. These tests are described in the following sections.

5.3.1 The *Wizard of UX* in .NET Code Camp

The second phase of the testing procedure was conducted in 2011 in a .NET Code Camp arranged in LUT (Lappeenranta University of Technology) [25]. Code Camps are intensive learning situations based on collaborative learning where the intention is to design and implement a working program within a given timeline, which usually is 24-72 hours. Code Camps have been arranged in our lab for approximately five years now and students seem to enjoy participating in them. During a Code Camp, students write programs together, solve problems related to their work together, eat together and might even relax together in a sauna [25]. The normal participant in a Code Camp is a 20-30 year old male who is interested in programming. Therefore, this course offered an excellent possibility to test the *Wizard of UX* with independent software developers who do not have knowledge about design for UX. 2011 .NET Code Camp course had 24 active participants who all were asked to use and evaluate the *Wizard of UX*. Thirteen responses were received, but four responses had to be removed due to inappropriate

values, e.g. empty row or average distribution of zero. Therefore, final analyses included nine respondents, which all were males. Only one respondent was outside of normal participant in age. Since the group of respondents is so homogeneous and small, the effect of gender, age or ability to use computers was not studied. This survey is presented in Appendix III.

The most important part of the survey was a series of statements concerning the usage and benefits of the *Wizard of UX*. Respondents were asked to evaluate statements with a seven-step likert scale. Numbers were not visible to the respondents, but are used in below calculations.

- Completely agree (1)
- Moderately agree (2)
- Slightly agree (3)
- Neither (4)
- Slightly disagree (5)
- Moderately disagree (6)
- Completely disagree (7)

In addition, the respondents had a possibility to answer 'I do not know', but this opinion was used only by one respondent in statement number 3. Table 12 presents the statements included in the survey with calculated statistical values of; mean (Mean), 95 % confidence interval (95 % CI), standard deviation (Std. Dev.) and variance (Var.).

Table 12: *Wizard of UX* evaluated by nine (9) software developers

Statement	Mean	95 % CI	Std. Dev.	Var.
1. I could increase my know-how about user experience by using the *Wizard of UX*	2,116	1,48-2,75	,830	,689
2. I could raise my understanding about end user(s) viewpoint by using the *Wizard of UX*	1,977	1,45-2,50	,735	,540
3. End user demographics have impact to UX	1,889	1,18-2,60	,928	,861
4. Some baseline(s) for my development project could be found by using the *Wizard of UX*	2,373	1,68-3,07	,905	,818
5. The *Wizard of UX* could aid me in my work	2,285	1,46-3,11	1,073	1,151
6. Current functionality of the *Wizard of UX* fills my needs	3,366	2,13-4,60	1,611	2,596
7. The *Wizard of UX* could be a useful tool for me	2,778	1,94-3,62	1,093	1,194
8. I would recommend the *Wizard of UX* to my UX friends or my UX oriented co-workers	2,793	2,04-3,54	1,047	1,097

All questions produced a mean value that was better than neutral (4) and even with 95 % confidence interval all expect one remained under the level of neutral. Confidence intervals are relatively large due to small amount of respondents. These results, in spite only nine respondents, show the potential of the *Wizard of UX*, for this group of software developers. The only statement that produced an average close to neutral with a big standard deviation was the question about the current functionality of the *Wizard of UX*. The results can be considered as encouraging since the *Wizard of UX* is just a first published prototype with very limited features.

In addition to responding the survey, there was a possibility to give free comments. As the number of respondents small, only two comments were given.

- *"It could be very good idea to create a web site to place all information about UX in one place. Surveys, articles, guidelines, user/professional grades..."*

- *"Could add some reasons for its scores as the scores itself are too vague to show the reason user or professionals would come to such a conclusion. I would also rather use words rather than score to show how much each subject is wanted by users or professional again because scores seems to vague for such thing."*

The first comment is a really valuable one and would greatly aid future UX research. The site *All About UX* (footnote 1 page 1) is partly targeting for this goal. However, in reality, it would be extremely difficult to collect all possible information in one place since the amount of knowledge is continuously increasing and UX research is distributed all around the world. In addition, part of the research is done by commercial companies who naturally may not be so willing to share their competitive advantage.

The second comment is true since numbers are used to give an impression of an absolute truth, which is a difficult one when target of the truth is subjective. Still use of numbers or words is a matter of personal preference and for this prototype numbers were chosen. In addition, some comments from UX experts about every included UX attribute would have been a good addition for the *Wizard of UX*.

Current evaluation only contains nine respondents so the results are not statistically valid. However, a trend can be red from the results presented in Table 12. Respondents moderately agreed on statements one and two, which indicates that the *Wizard of UX* is apparently a potential tool for increasing software developers' know-how about user experience as well as raising understanding about end user(s) viewpoint. This was the first objective O1 (*Raising the UX knowledge among software developers.)* set for the *Wizard of UX* prototype. Therefore it can be stated that according to these results the first objective was met.

This group of respondents also moderately agreed that end user demographics have impact to UX (statement 3 in Table 12). Furthermore, respondents also moderately agreed that the *Wizard of UX* could aid them in their work (statement 5, in Table 12) and could offer some baselines for development projects (statement 4 in Table 12). When these statements are considered with slight agreement on statement 7 (The

Wizard of UX could be a useful tool for me), it can be stated that the second objective O2 (*Considering the attributes of UX by utilizing statistical user profiles in the early phases of software development.)* and third objective O3 (*To prototype and test a utility that assists in meeting O1 and O2.)* was met

5.3.2 UX designer evaluation

The third and final phase was to evaluate the *Wizard of UX* with industrial UX professional. This test was conducted with UX expert evaluation method by utilizing a cognitive walk-through. UX designer Janne Romppainen from Tieto who has more than five year experience in designing for various clients and tasks conducted the evaluation. In his opinion the *Wizard of UX* is suitable for conducting base study concerning the chosen target users before user interviews to direct interview questions into right direction. Most useful part based on his experience, for software developers without previous UX knowledge, would be the part explaining different UX attributes and giving suggestions how to utilize UX attributes in software development.

Moreover, he gave suggestions how to improve the functionality on the *Wizard of UX*. Firstly, an introduction page with explanation and links to functionality would be beneficial. Secondly, bars presenting the results should be presented so that larger is better and thirdly some middle headers for the results page is needed in order to clarify the view. Fourthly, additional information from the results should be given with the form of text. His final conclusion of the *Wizard of UX* was that the additional information from the *Wizard of UX* is not enough to reduce the amount of conducted interviews in industrial context before the functionality has been improved and validated but for a software developer without UX experience this utility is a good way to start considering UX.

In addition, discussions were concerned of how user surveys are done in industrial context. Based on his experience of clients and software developers, they do not have time to answer multiple questions concerning same issue with different wordings. When time is precious, simplicity and direct questions are appreciated, as was done in the above case.

5.3.3 Redesigning the *Greenhouse*

In addition, to testing the benefits of using the *Wizard of UX* in the .NET Code Camp, the *Wizard of UX* was utilized for taking the firsts steps towards redesigning the mobile multi-user game called the *Greenhouse* (see Section 1.1). The *Greenhouse* was designed and implemented by the author who also was responsible for all features. In the original development, the end user viewpoint for the *Greenhouse* was never asked. Table 13 presents the starting points for the development of the original *Greenhouse* game five years ago. These starting points are presented according to the UX attributes in Section 4.5. Priority in the table goes from 1-3 where 1 is the most important one, 2 is nice, but not necessary, and 3 is not even considered. The right-hand side of the table gives an explanation for the chosen priority level.

Table 13: Starting points of the *Greenhouse*

UX attribute	Priority	Explanation
Accessibility	3	If one is able to use modern mobile phone that is capable of running MUPE, there is no need to consider accessibility issues.
Aesthetics	1	Was considered as one of the main attributes among the nice UI.
Brand	3	It was possible to run under any device supporting Java and MIDP (Mobile Information Device Profile) 2.0.
Consistency	2	Some things were forced to put on the action keys of the phone, but not particular accuracy were given to consistency of soft buttons.
Coolness	1	Intention of virtually every game.
Device	1	Features and capabilities of phone placed lots of restriction to the design. E.g. screen resolution and amount of memory.
Ease of taking into use	2	Was handled by the MUPE client, only thing required was to download the *Greenhouse* into the MUPE client and start using it.
Emotions	3	I did not have advance information, that end user emotions might have effect to design a game.
Environment	3	I did not have advance information, that use environment might have effect to design a game.
Expectations	3	I did not have advance information, that end user expectations might have effect to design a game.
Functionality	1	Intention was to simulate real greenhouse and marketplace so the functionality was one of the main points.
Interaction	2	Was not added until the last meters of the development, due to deadline.
Motivation	3	I did not have advance information, that motivation of end user might have something to do with design a game.
Pleasure	1	One of the main points of game is to offer pleasure for the player.
Price	2	At this time there were not 3G data packages so the amount of transferred data was charged from the player. Therefore, MUPE offered a data compression that the *Greenhouse* supported.
Reliability	2	In a case of problems, it was almost always required to clear everything from the server and reset the client. Problems were tried to identify but new features and nice graphics went over these 'secondary' goals.
Stability	2	*Greenhouse* was not known for its stability, attempts were made to fix this, but issues remained.
Trust & Privacy	3	User profile was saved to personal device but it was not even considered that someone might do something nasty with it.
Usability	2	Intention was to make it usable but again this was just a secondary goal.
Usefulness	3	Games in generally are not useful.
User interface	1	Among aesthetics one of key attributes that I was concerned about.

In redesigning process first the potential end users of the *Greenhouse* game were identified. Females were not seen as active in playing games as males, and therefore, the primary target group was restricted to males. Many people play games, but it was assumed that the most active gamers are somewhere between 15-24 years old. Finally,

people who are not comfortable with using technological devices are very unlikely to play mobile games and people who, e.g. design and develop mobile games for living would probably not like to use games in their free time. Therefore, the ability to use computers was defined as from 3-5. These values were fed into the *Wizard of UX* and the results are shown in Figure 15.

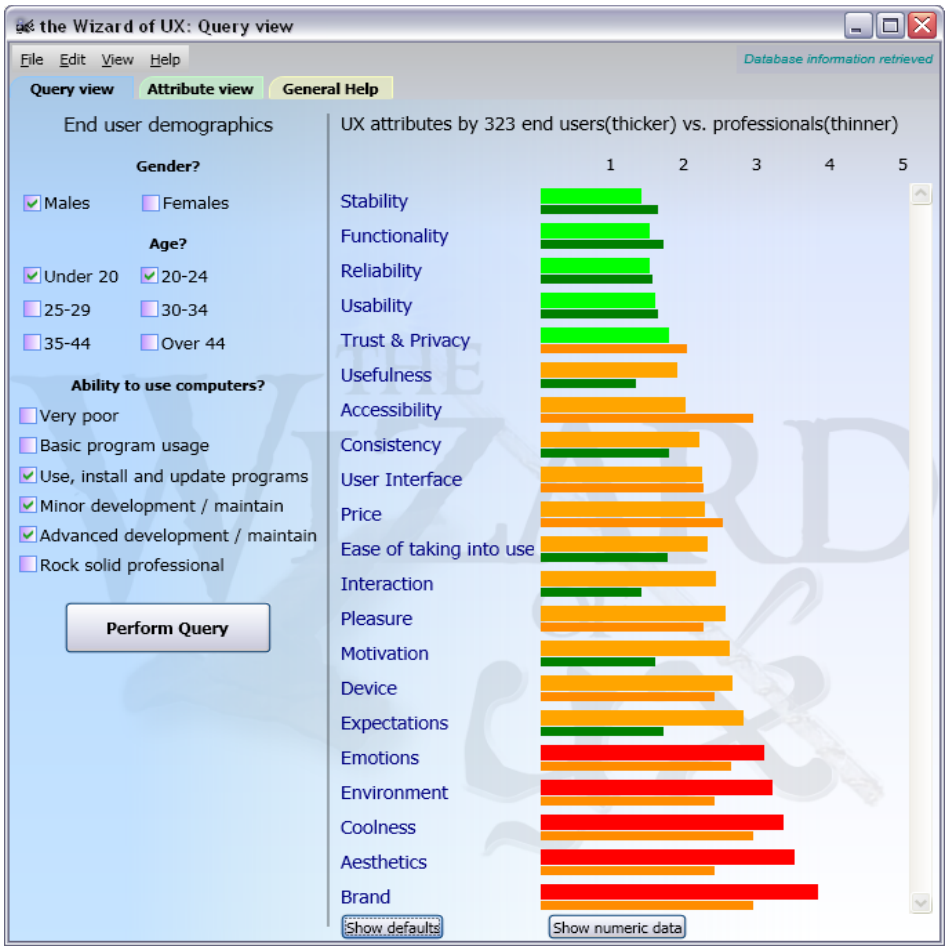


Figure 15: Results from the *Greenhouse* target group

The results show that some of the (un)sophisticated guesses made during the MUPE project were in fact correct and that it was a valid decision to focus on functionality, ignoring attributes like expectations, emotions, environment, and brand. Based on the starting points for the *Greenhouse* given by the *Wizard of UX*, a lot more attention would have been given to stability, reliability, usability and trust and privacy issues. Moreover, once these attributes had been adequately addressed, attention would have been given attention to UI and its consistency, as well as making the *Greenhouse* more

accessible. On the other hand, the aim of trying to make the *Greenhouse* as cool as possible and aesthetically pleasing would not have been pursued.

5.3.4 Redesigning the *Activator*

Currently, the *Wizard of UX* does not take into account whether users in the target group happen to suffer from some disability. Therefore, the current possibilities offered by the *Wizard of UX* are not suitable for software design for people with disabilities. An attempt was made to address this shortcoming by using a combination of mental and physical ability to use computers (levels 1-3). Figure 16 presents the results given by the *Wizard of UX* in numerical format.

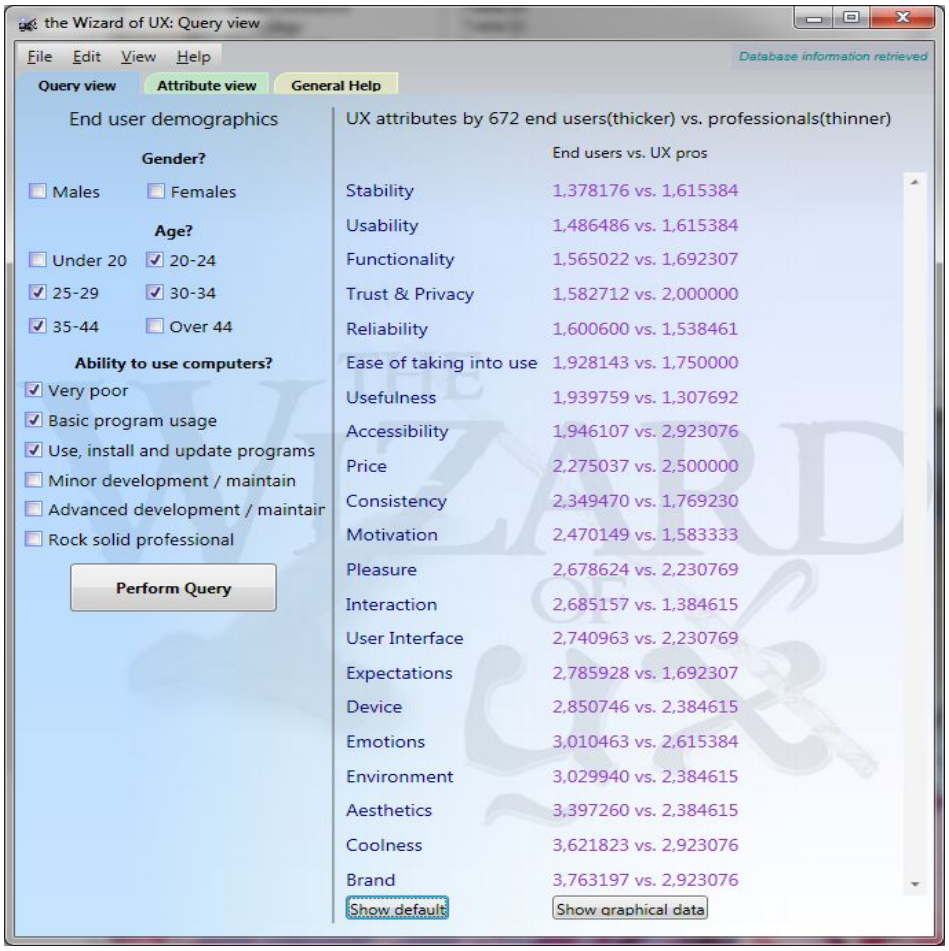


Figure 16: *Wizard of UX* - *Activator* target group

The *Activator* was designed to be used by MS patients and to offer enjoyable moments together, brain teasers and physical movement. Originally the *Activator* was designed in

co-operation with occupational therapists so, in theory, the abilities and wishes of end users were known. Unfortunately, the main focus of occupational therapists was on the content of the *Activator* and not how items should be presented, how accessibility should be guaranteed, etc. Therefore, the actual design and implementation were based on authors' considerations and judgments. After initial contact with the target users, many suggestions were put forward, like a possibility to enlarge fonts (accessibility), lowering the detail level of the map (usability, usefulness) and requests for new features such as a bulletin board, real time conversation and the inclusion of important locations (functionality, usefulness).

Persons suffering from MS disease cannot be categorized according to gender; however, age is a factor as symptoms of the disease usually begin around the age of 20-40. Categorization according to ability to use computers is difficult since a person suffering from MS disease might have been, e.g. a programmer before the disease, but the symptoms might have made it physically impossible to use computers.

The results gained from the *Wizard of UX* are well in line with the feedback from the initial contact (accessibility, usability, usefulness, functionality) despite the fact that the current database does not contain information about disabilities. With hindsight and based on the above considerations, more attention would have been paid to stability, reliability and trust and privacy issues and focused less on polished UI and aesthetics.

5.4 Related solutions

Solutions for UX considerations in the early phases of software development presented in the site *All About UX* (footnote 1 page 1) are mostly based on interviews, recordings, evaluations, etc., and the only concrete tool is Kansei Engineering Software [145], presented in sub section 3.8.1. Existing methods are either unsuitable for the target area, require a possibility to use outside subjects such as end users or UX professionals or are usable only after the target area when something concrete already exists. Therefore the Wizard of UX is new and unique utility for integrating UX into early phases of software development.

5.5 Discussion

This chapter aims to respond directly to Q (*How to enhance UX considerations in the early phases of software development without big investments?)* by presenting the *Wizard of UX* utility, which is easy to take into use and free. According to conducted tests (see Table 12), the prototyped utility was seen as a useful tool and a beneficial way in increasing user experience know-how among software developers during the early phases of software development. When the general UX knowledge is raised, also the ability to consider UX is enhanced. Chapter described the process of designing, prototyping and testing the *Wizard of UX*. Based on the conducted tests; .NET Code Camp, UX designer evaluation, and two test cases, the prototyped *Wizard of UX* seems to fulfill the research objectives of this dissertation:

- O1: *Raising the UX knowledge among software developers.*
- O2: *Considering the attributes of UX by utilizing statistical user profiles in the early phases of software development.*
- O3: *To prototype and test a utility that assists in meeting O1 and O2.*

.NET Code Camp participants moderately agreed that they could increase their know-how about UX and raise their understanding about end user(s) viewpoint with the *Wizard of UX*. UX designer considered that the *Wizard of UX* might be a good way to start considering UX if previous knowledge about UX is missing. Furthermore, both redesigning cases showed that if the end user's perspective on UX is not considered in early phases of development, modifications are required afterward. If *Wizard of UX* would have been used for these cases, at least some of these modifications would have been avoided.

6 Conclusion and future work

In this dissertation the author has claimed that "*The Wizard of UX utility increases software developers' know-how, understanding and ability to consider UX by providing UX information and target users' view on the importance of UX attributes.*". To prove the claim, the current trends of UX research and the status of UX in software development was presented. It seems that the tentacles of UX, experience, an experience, co-experience, etc. have reached virtually everything that surrounds us. Its power can be seen in various channels like the travel industry, commercials, news, and software development, but the question remains; what is UX and from whose point of view?

The scope of this dissertation was limited to considering UX in software development, in which the existing methods are biased to the latter end of the development life cycle. Still it has been a long-term wish, that UX could be evaluated and analysed during the early phases of software development [149]. Therefore a need for a new utility was evident. Research question: "*How to enhance UX considerations in the early phases of software development without big investments?"* and three objectives; O1 (*Raising the UX knowledge among software developers.), O2 (Considering the attributes of UX by utilizing statistical user profiles in the early phases of software development)* and O3 *(To prototype and test a utility that assists in meeting O1 and O2.)* of this dissertation were set to support the evident need.

The practical part of this dissertation, which starts from Chapter 4, describes the design and process of conducting UX surveys as well as identification of the included UX attributes. Two surveys were conducted, UX professional survey and end user survey. Both surveys consist of background information, UX definitions and UX attributes. Background information refers to gender, age and ability to use computers and was asked to be able to categorize respondents. UX definition part is not included in the prototyped UX utility but its intention was to demonstrate the apparent trend that; an average end user of software has little interest in UX definitions. According to results it seems that 25 % of end users had no interest in defining user experience and responses from this end user sample were scattered among every given definition. UX professionals apparently have two preferred definitions. Last part of the surveys was UX attributes. For the scope of this dissertation, UX was demarcated in to chosen 21 attributes, which were considered to be comprehensive enough for the purpose of this dissertation. However a notification must be given; in spite demarcation of UX being suggested [135], it needs to be question whether important information was lost due to

this action. The final results of surveys include 12 UX professional respondents and 1420 end user respondents. The used end user sampling is too uniform for widespread generalization, but according to results, statistically significant dependencies between perspective towards UX attributes and personal demographics and UX attributes and ability to use computers were found. From the surveyed 21 UX attributes, four (coolness, ease of taking into use, interaction and user interface) seemed to be dependent on every surveyed background information.

Finally, utilization of the results in designing, prototyping and testing of the *Wizard of UX* utility is described in Chapter 5. The *Wizard of UX* was designed to be used in early phases of software development and provides an end user perspective towards importance of UX attributes. In addition, end user perspective is compared against personal appreciations of UX professionals. Explanations for UX attributes and hints for software development are also part of the *Wizard of UX* utility. Prototyped *Wizard of UX* was tested with a group of nine software developers participating in an intensive programming oriented course and with an industrial UX designer. Developers saw the potentiality of this utility for increasing the know-how and understanding of UX and UX designer evaluated the *Wizard of UX* as a good way for a software developer without previous UX knowledge, to start with. In addition, two reference applications were redesigned by utilizing the knowledge offered by the *Wizard of UX*. In both redesigning cases, some modifications required during the latter parts of development would have been avoided if the *Wizard of UX* would have been available in the early phases. According to the received results, the *Wizard of UX* is potential and useful solution for considering attributes of UX in early phases of software development.

This dissertation is a response to the set research question: *How to enhance UX considerations in the early phases of software development without big investments?* Designed and prototyped user experience utility is free and simple to use and according to the tests conducted by software developers and UX designer, it was seen as a potential way in increasing user experience know-how and ability to consider UX during early phases of software development. Three objectives set for this dissertation were also met: The UX knowledge among software developers, according to result can be raised, statistical user profiles and attributes of UX can be taken into account during the early phases of software development with the *Wizard of UX* utility, and finally the prototyped *Wizard of UX* is the response to the objective number three. As a response to the claim, target users' view on the importance of UX attributes was collected and according to the results this information seems to be usable in increasing software developers' knowledge of UX with the prototyped *Wizard of UX* utility.

In spite this dissertation produces new and relevant information for the UX and software development communities as well as makes it possible to integrate UX into the early phase of software development, it has its limitations: 1. Results are based on samplings, so widespread generalization cannot be done. 2. Used list of 21 UX attributes is far from complete, but universally accepted UX attributes for evaluating and analysing UX have not been identified. In addition, scientific method for selecting the 21 UX attributes was not used, instead attributes were collected from various sources. 3. *Wizard of UX* is currently just a prototype and does not consider for example end users with some

physical or mental disabilities, or the type of development target (game, utility, leisure, work etc.).

Possible future work can be tied to presented limitations: 1. More information on the importance of UX attributes to end users is needed to be able to clarify further the effect of demographics and different abilities. This knowledge could be harvested by utilizing Facebook[35] messages, online communities or big institutions like schools or workplaces. In addition, it would be beneficial if the *Wizard of UX* could reach people with disabilities, children, elderly people, and etc. 2. UX attribute list could be enhanced by making it more thorough and by studying correlations between attributes. 3. Improvements to the *Wizard of UX* are required to make it work in practice. Currently, the *Wizard of UX* utility only offers basic functionality. More studies about the benefits of using the *Wizard of UX* in industrial software development are also needed. Lastly, the *Wizard of UX* could be tested in a real context like an industrial development project to be able to see the long-term benefits of utilizing it in the early phases of the software development.

To finally conclude this dissertation the author presents the following partial and modified (The Matrix → User experience) quote by Morpheus in the movie Matrix[36] which suits UX well: "*User experience is everywhere. It is all around us. Even now, in this very room...*". It is authors' sincere hope is that the increase in knowledge and know-how of UX provided by this dissertation and the *Wizard of UX* utility will aid software developers to take UX of end users into account as early as possible during software development project.

35 http://www.facebook.com/
36 http://www.thematrix101.com/

References

[1] ABC News, Computer error behind Qantas midair drama" Online supplement, Available http://www.abc.net.au/news/stories/2008/10/14/2391134.htm 3.1.2011

[2] Agile Usability Group, Online supplement, Available http://tech.groups.yahoo.com/group/agile-usability/ 5.1.2011

[3] Alakärppä, I. Riekki, J. Koukkula, R. Pervasive Pain Monitoring System: User Experiences and Adoption Requirements in the Hospital and Home Environments. In proceedings of Pervasive Health'09, (London, UK, 2009)

[4] Alben, L. At the Heart of Interaction Design. Design management journal, 8, 3, (1997), 9-26

[5] Alben, L. Quality of experience: Defining the criteria for effective interaction design. ACM interactions, May/June, (1996), 11-17.

[6] All About UX – Information for user experience professionals. Online supplement. Available http://www.allaboutux.org/ 9.2.2011

[7] Amaral, J. R. Oliveira, J. M. Limbic System: The Center of Emotions.Online supplement, available http://www.healing-arts.org/n-r-limbic.htm 22.9.2011

[8] Andreasen, M. Nielsen, H. Schrøder, S. Stage, J. Usability in Open Source Software Development: Opinions and Practice. Information Technology and Control 25,3A,(2006), 303-312.

[9] Arhippainen, L. Studying User Experience: Issues and Problems of Mobile Services – Case ADAMOS: User Experience (im)Possible to Catch? Doctoral dissertation. University of Oulu. 2009.

[10] Ask Bob Rankin. Online supplement, available http://askbobrankin.com/worst_problem_on_the_net_stupid_users.html, 6.5.2011

[11] Au, I. Boardman, R. Jeffries, R. Larvie, P. Pavase, A. Riegelsberger, J. Rodden, K. Stevens, M. User Experience at Google – Focus on the user and all else will follow. In proceedings of CHI 2008, (Florence, Italy, 2008), 3681-3686

[12] Bad Human Factors Designs, Online supplement, Available http://www.baddesigns.com/examples.html 13.1.2011

[13] Bannerman, P. L. Risk and risk management in software projects: A reassessment, Journal of systems and software, 81, 12, (2008), 2118-2133.

[14] Bargas-Avila, J. A. Lötscher, J. Orsini, S. Opwis. K. Intranet satisfaction questionnaire: Development and validation of questionnaire to measure user satisfaction with the Intranet. Computers in Human Behavior, 25, ,(2009), 1241-1250.

[15] Beck, K. Extreme Programming Explained: Embrace Change. Addison-Wesley, 1999, ISBN: 978-0201616415

[16] Bellmore, A. Villarreal, V. M. Ho, A. Y. Staying Cool Acress the First Year of Middle School. Journal of Youth and Adolescence, 39, (2010), 1-10.

[17] Berkum, S. How to Avoid Foolish Consistency, MSDN, Online supplement, Available http://msdn.microsoft.com/en-us/library/ms993287.aspx 15.2.2011

[18] Bevan, N. Classifying and Selecting UX and Usability Measurements. In proceedings of VUUM 2008, (Reykjavik, Iceland, 2008).

[19] Budgen, D. Software Design, second edition. Pearson Education Limited, 2003, ISBN: 0-201-72219-4

[20] Budwig, M. Jeong, S. Kelkar, K. When User Experience Met Agile: A Case Study, In proceedings of CHI 2009, (Boston, MA, USA, 2009), 3075-3084

[21] Bullinger, H-J. Bauer, W. Wenzel, G. Blach, R. Toward user centred design (UCD) in architecture based on immersive virtual environments. Computers in Industry, 61, 4, (2010), 372-379

[22] Buxton, B. Sketching User Experience: Getting the Design Right and the Right Design. Morgan Kaufman, 2007, ISBN: 0-1237-4037-1

[23] Chou, C. Hsiao, M-C. Internet addiction, usage, gratification, and pleasure experience: the Taiwan college student's case. Computers & Education, 35,1, (65-80), 2000

[24] Clark. J. Building Accessible Websites. New Riders, 2002, ISBN: 0-7357-1150-X

[25] Codecamp, www-page. Online supplement, Available http://www.codecamp.fi/ 28.3.2011

[26] Cooper, A. The inmates are running the asylum. Sams Publishing, 2004, ISBN:0-672-32614-0

[27] Da Silva, T. S. Martin, A. Maurer, F. Silveira, M. User-Centered Design and Agile Methods: A Systematic Review. In proceeding of Agile 2011, (Salt Lake City, Utah, USA, 2011), 77-86

[28] De Angeli, A. Hartmann, J. Sutcliffe, A. The Effect of Brand on the Evaluation of Websites. In proceedings of Interact'09, (Uppsala, Sweden, 2009),

[29] Deci, E. L. Ryan, R. M. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being, American Psychologists, 55, (2000), 68-78.

[30] Demarcating User eXperience seminar, Online supplement, Available http://www.dagstuhl.de/en/program/calendar/semhp/?semnr=10373, 8.2.2011

[31] Desmet, P. Hekkert, P. Framework of Product Experience, International Journal of Design, 1, 1, (2007), 57-66.

[32] Desmet, P. M. A. Overbeeke, C. J. Tax, S. J. E. T. Designing products with added emotional value; development and application of an approach for research through design. The Design Journal, 4, 1, (2001), 32-47.

[33] Dewey, J. "Art as Experience", Perigee Trade, 1980, ISBN:0-3995-0025-1

[34] Dewey, J. "Experience and Education", Free Press, 1997 (1938), ISBN:0-6848-3828-1

[35] Diederich, M. Sex Differences in Experiences and Career Plans of Freshman Engineering Students. IEEE Transactions on Education, 21, 4, (1978), 230-233

[36] DOD-STD-2167, Military Standard: Defense System Software Development, Online supplement, Available: http://www.product-life cycle-management.com/download/DOD-STD-2167A.pdf 5.1.2011

[37] Dror, I. A holistic-cognitive approach for success. Biometric Technology Today, 14, 7-8, (2006), 7-8.

[38] Dumas, J. S. Redish, J. C. A Practical Guide to Usability Testing. Intellect, 1999. ISBN: 978-1841500201

[39] Editorial note. The year 2000 Final Though – Stupid is as Stupid Does! Computer Fraud & Security Journal, 12, 1, (2000), 17-19

[40] Eriksson, K. Lindholm, T. Making gender matter: The role of gander-based expectancies and gender identification on women's and men's math performance in Sweden. Scandinavian Journal of Psychology, 48, (2007).

[41] Etchemendy, E. Baños, R. M. Botella, C. Castilla, D. Alcañiz, M. Rasal, P. Farfallini, L. An e-health platform for the elderly population: The butler system. Computers & Education, 56, 1, (2011), 275-279.

[42] Fallman, D. Waterworth, J. Dealing with User Experience and Affective Evaluation in HCI Design: A Repertory Grid Approach. In proceedings of CHI 2005, (Portland, Oregon, US, 2005).

[43] Fayad, M. E. Altman, A. Thinking objectively: an Introduction to Software Stability. Communications of the ACM, 44, 9, (2001), 95-98

[44] Federoff, M. Courage, C. Successful User Experience in an Agile Enterprise Environment. In proceedings of HCII 2009, (San Diego, CA, USA, 2009), 233-242

[45] Fenton, N. E. Pfleeger, S. L. Software Metrics: A Rigorous & Practical Approach. PWS Publishing Company, 1997, ISBN: 0-534-95425-1

[46] Finnish MS-society, www-pages. Online supplement. Available http://www.ms-liitto.fi/english 14.3.2011

[47] Forlizzi, J. Battarbee, K. Understandin Experience in Interactive Systems. In proceedings of DIS 2004, (Boston, USA, 2004), 261-268

[48] Forlizzi, J. Ford, S. The Building Blocks of Experience: An Early Framework for Interaction Designers. In Proceedings of DIS '00 (Brooklyn, NY, USA, 2000) pp.419-423

[49] Følstad, A. The Relevance of UX models and Measures. In proceedings of I-UxSED 2010, (Reykjavik, Iceland, 2010).

[50] Gaffney, G. Fouder of Information & Design company. The myth of the stupid user, Online supplement, Available http://www.infodesign.com.au/articles/themythofthestupiduser 3.1.2011

[51] Goble, F. G. The Third Force: The Psychology of Abraham Maslow, Viking Adult, 1970, ISBN: 0-6707-0065-7

[52] Green, T. R. G. The Cognitive Dimensions of Viscosity: a Sticky Problem for HCI. Human-Computer Interaction, 90, (1990), 79-86.

[53] Haikala, I. Märijärvi, J. Ohjelmistotuotanto. Talentum, 2006. ISBN: 952-14-0850-2

[54] Harris, M. Windows Mobile phones suffer Y2K+10 bug, World of tech news, 3.1.2010, online supplement, Available http://www.techradar.com/news/world-of-tech/windows-mobile-phones-suffer-y2k-10-bug-661062 10.1.2011

[55] Hassenzahl, M and Tractinsky N. User Experience – a research agenda. Behaviour & Information Technology journal 25, 2, (2006), 91-97

[56] Hassenzahl, M. Burmester, M. Koller, F. AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität. Mensch & Computer, 2003, 187-196.

[57] Hassenzahl, M. Schöbel, M. Trautman, T. How motivational orientation influences the evaluation and choise of hedonic and pragmatic interactive products: The role of regulatory focus. Interacting with computers, 20, 4-5, (2008), 473-479

[58] Hassenzahl, M. The thing and I: Understanding the Relationship Between User and a Product. In Funology: From Usability to Enjoyment. Kluwer Academic Publishers, 2003, ISBN: 1-4020-2966-7

[59] Hassenzahl, M. User Experience (UX): towards an experiential perspective on product quality. In proceedings of IHM 2008, (Metz, France, 2008), 11-15

[60] Hedberg, H. Iivari, N. Integrating HCI Specialists into Open Source Software Development Projects. In proceedings of OSS 2009, (Skövde, Sweden, 2009), 251-263

[61] Henry, J. Software Project Management. Addison-Wesley, 2003, ISBN: 0-2017-5865-2

[62] Hentunen, M. Äiti, paljon sun tilillä on rahaa?. Lapsemme, 4, (2010), 8-13. Available http://www.mll.fi/@Bin/13199237/Lapsemme4_2010netti.pdf (in Finnish)

[63] Hsu, S. H. Wen, M-H. Exploring user experiences as predictors of MMORPG addiction. Journal of Computers & Education, 53, 3, (2009), 990-999

[64] Hudson, J. M. Kameshwari, V. Feature – Can "wow" be a design goal? Interactions, 16, 1, (2009), 58-61.

[65] IEEE standard 610.12-1990.

[66] Iivari, J. A paradigmatic analysis of contemporary schools of IS development. European Journal of Information Systems, 1, 4, (1991), 249-272

[67] Instone, K. User experience: an umbrella topic. In proceedings of CHI'05, (Portland, Oregon, USA, 2005), 1087-1088

[68] Iqbal, M. Rizwan, M. Application of 80/20 rule in software engineering Waterfall Model. In proceedings of ICICT'09, (Karachi, Pakistan, 2009), 223-228

[69] ISO 9241-11:1998 Guidance on Usability

[70] ISO 9241-210:2010. Ergonomics of human-system interaction –part 210: Human-centred design for interactive systems. 2010

[71] Itkonen, J. Mäntylä, M. V. Lassenius, C. How Do Testers Do It? An Exploratory Study on Manual Testing Practices. In proceedings of ESEM'09, (Florida, USA), 494-497.

[72] Jalote, P. A Concise Introduction to Software Engineering. Springer-Verlag, 2008, ISBN: 978-1-848-00301-9

[73] Jokela, T. Characterizations, Requirements and Activities of User-Centered Design-the KESSU 2.2 Model. in Maturing Usability, Springer-Verlag, 2008, ISBN:978-1-84628-940-8

[74] Jokela, T. Determining Usability Requirements into a Call-for-Tenders. A Case Study on the Development of a Healthcare System. In proceeding of NordiCHI 2010, (Reykjavik, Iceland, 2010)

[75] Jokela, T. Joticon Oy. Online supplement, Available http://iso9241-210.blogspot.com, 15.12.2010

[76] Jones, S. R. G. Was There a Hawthorne Effect? American Journal of Sociology, 98, 3, (1992), 451-468.

[77] Jordan, P. W. Designing Pleasurable Products. CRC Press, 2002, ISBN:978-0415298872

[78] Joshi, A. Sarda, N. L. HCI and SE: towards a 'truly' unified waterfall process. In proceedings of HCI International 2007, (Beijing, China, 2007)

[79] Joshi, A. Sarda, N. L. Tripathi, S. Measuring effectiveness of HCI integration in software development processes. Journal of Systems and Software, 83, 11, (2010), 2045-2058

[80] Jumisko-Pyykkö, S. Vainio. T. Framing the Context of Use for Mobile HCI. International Journal of Mobile Human Computer Interaction, 2, 4, (2010), 1-28.

[81] Jääskeläinen, A. Heikkinen, K. Designing, Implementing and Testing Experiental Multi-User Virtual Community With MS-Patients. International Journal on WWW/Internet 5, (2007), 147-164

[82] Jääskeläinen, A. Heikkinen, K. Divergence of User eXperience: Professionals vs. End Users. In proceedings of I-UxSED 2010, (Reykjavik, Iceland, 2010).

[83] Jääskeläinen, A. Lautamäki, J. Analysing Context-Aware Service Development under MUPE Platform. In Proceedings of ASWN '08 (Kassel, Germany, 2008) pp. 26-34

[84] Jääskeläinen, A. User eXperience: Tool for developers. In proceedings of Interact 2009, (Uppsala, Sweden, 2009), 888-891.

[85] Kaner, C. Pels, D. Bad Software: what to do when software fails. Wiley, 1998, ISBN: 0-4713-1826-4

[86] Kansei Engineering, Online supplement. Available http://www.kansei.eu/, 10.2.2011

[87] Karapanos, E. Martens, J.-B. Hassenzahl, M. On the retrospective assessment of users' experiences over time: memory of actuality? In proceedings of CHI CHI'10 (Atlanta, GA, USA, 2010), 4075-4080.

[88] Karapanos, E. Zimmerman, J. Forlizzi, J. Martens, J-B. User experience over time: an initial framework. In proceedings of CHI 2009, (Boston, MA, USA, 2009), 729–738.

[89] Kolko. J. Thoughts on Interaction Design: A Collection of Reflections. Morgan Kaufman, 2011. ISBN: 0-1238-0930-4

[90] Kollmann, J. Sharp, H. Blandford, A. The importance of Identity and Vision to user experience designers on agile projects. In proceedings of AGILE'09, (Chicago, USA, 2009), 11-18

[91] Korhonen, H. Koivisto, E. M. I. Playability Heuristics for Mobile Games. In proceeding of MobileHCI' 06, (Espoo, Finland, 2006), 9-16.

[92] Kuhn, T. S. The function of measurement in modern physical science. in Quantification. A history of the meaning of measurement in the natural and social sciences. Bobbs-Merrill, 1961.

[93] Kujala, S. Roto, V. Väänänen-Vainio-Mattila, K. Karapanos, E. Sinnelä, A. UX Curve: A method for evaluating long-term user experience. Interacting with computers (in press), (2011)

[94] Laerd statistics, Pearson Product-Moment Correlation, Online supplement. available http://statistics.laerd.com/statistical-guides/pearson-correlation-coefficient-statistical-guide.php, 11.3.2011

[95] Lavie, T. Tractinsky, N. Accessing dimensions of perceived visual aesthetics of web sites. Human-Computer Studies, 60, (2004), 269-298

[96] Law, E. L-C, Roto, V. Hassenzahl, M. Vermeeren, A. P.O.S. Kort, J. Understanding, Scoping and Defining User eXperience: A Survey Approach. In Proceedings of CHI 2009 (Boston, MA, USA, 2009) pp. 719-728

[97] Law, E. L-C. Schaik, P. Modelling user experience – An agenda for research and practice. Interacting with computers, 22, 5, (2010), 313-322.

[98] Lederer, S. Hong, I. Dey, K. Landay, A. Personal privacy through understanding and action: five pitfalls for designers. Personal and ubiquitous Computing, 8, 6, (2004), 440-454.

[99] Leicht, M. Managing User Expectations. Online supplement, Available http://www.umsl.edu/~sauterv/analysis/user_expectations.html 21.2.2011

[100] Lott, C. M. Breathing New Life into the Waterfall Model. Software Journal, 14, 5, (1997), 103-105

[101] Luojus, S. From a momentary experience to a lasting one. Doctoral dissertation. University of Oulu. 2010.

[102] Lyu, M. R. Software Reliability Engineering: A Roadmap. In proceedings of FOSE'07, (Minneapolis, USA, 2007), 153-170.

[103] Löwgren, J. Stolterman, E. Thoughtful Interaction Design: A Design Perspective on Information Technology. MIT Press, 2004, ISBN: 0-2621-2271-5

[104] Mahdy, A. Fayad, M. E. A Software Stability Model Pattern, In proceedings of PLOP02, (Monticello, Illinois, US)

[105] Mason, J. Qualitative Researching, Sage Publications Ltd, 2002, ISBN: 978-0761974284

[106] Meister, D. The History of Human Factors and Ergonomics. CRC Press, 1999, ISBN: 0-8058-2769-2

[107] Merriam-Webster Online dictionary. Online supplement, Available http://www.merriam-webster.com/, 21.9.2011

[108] Microsoft. Design: A Changing Culture. Online supplement. Available http://www.microsoft.com/design/article.aspx?type=culture&key=design 13.1.2011

[109] Miller-Cochran, S. K. Rodrigo, R. L. The Wadsworth Guide to Research. Wadworth, 2009, ISBN: 0-4957-9966-1

[110] Molokken-Ostvold, K. Jorgensen, M. A Comparison of Software Project Overruns-Flexible versus Sequential Development Models. Transactions of Software Engineering, 31, 9, (2005), 754-766

[111] Moshagen, M. Thielsch, M. T. Facets of visual aesthetics. International Journal of Human-Computer Studies, 68, 10, (2010), 689-709.

[112] Moss, G. Differences in the design aesthetics of men and woman: Implications for product branding. Journal of Brand Management, 3, , (1995), 51-61

[113] Moss, G. Gender and costumer behavior: Further explorations. Journal of Brand Management, 7, , (1999), 88-100.

[114] Moss, G. Horvath, G. Gunn, R. Refining our understanding of men and women's design preferences. In proceedings of Workshop on Towards a UX Manifesto 2007, Online supplement, Available http://141.115.28.2/cost294/upload/506.pdf 7.1.2011

[115] Myers, G. J. Sandler, C. Badgett, T. Thomas, T. M. The Art of Software Testing, Second Edition. Wiley & sons, 2004, ISBN: 0-4714-6912-2

[116] Mäkelä, A. Fulton Suri, J. Supporting Users' Creativity: Design to Induce Pleasurable Experiences. In proceedings of the International Conference on Affective Human Factors Design, (Singapore, 2001), 387-394

[117] Najafi, M. Toyoshiba, L. Two case studies of user experience design and agile development. In proceedings of AGILE'08, (Toronto, Canada, 2008), 531-536

[118] Naumann, A. B. Wechsung, I. Hurtienne, J. Multimodal interaction: A suitable strategy for including older users? Interacting with Computers, 22, 6, (2010), 465-474.

[119] Naur, P. Randell, B. Software Engineering: Report on a conference sponsored by the NATO Science Committee. (Garmisch, Germany, 1968)

[120] Neumann, P. G. Computer Related Risks. Addison Wesley, 1994, ISBN: 0-2015-5805-X

[121] Nichols, D. Twidale, M. Usability Processes in Open Source Projects. Software Process Improvement and Practice 11, (2006), 149-162.

[122] Nielsen Norman Group, User Experience – our Definition, Online supplement, Available http://www.nngroup.com/about/userexperience.html 14.2.2011.

[123] Nielsen, J. Usability engineering. Morgan Kaufmann, 1994, ISBN: 0-1251-8406-9

[124] Norman, D. A. The Psychology Of Everyday Things. Basic books, 1988, ISBN: 978-0465067091

[125] Norman. D. A. Emotional Design. Why we love (or hate) everyday things. Basic Books, 2003, ISBN: 0-4650-5135-9

[126] Novak, J. Schmidt, S. When Joy Matters: The Importance of Hedonic Stimulation in Collocated Collaboration with Large-Displays. Human-Computer Interaction, 57, 27, (2009), 618-629

[127] Nunan, D. Research Methods in Language Learning. Cambridge University Press, 1992, ISBN: 9780521429689

[128] Pearsall, P. Pleasure Prescription: To Love, to Work to Play – Life in Balance, Hunter house, 2000, ISBN: 0-8979-3207-2

[129] Pine, B. J. Gilmore, J. H. the Experience Economy: Work is Theatre & Everyday Business a Stage. Harvard Business School Press, 1999, ISBN: 0-8758-4819-2

[130] Rabbe, L. Post-mortem on the Skype outage. Skype blog, 29.12.2010. Online supplement. Available http://blogs.skype.com/en/2010/12/cio_update.html 11.1.2011

[131] Read, J. C. Validating the Fun Toolkit: an instrument for measuring children's opinions of technology. Cognition, Technology & Work, 10, 2, (2008), 119-128.

[132] Root, R. W. Draper, S. Questionnaires as a Software Evaluation Tool. In proceedings of CHI'83 (Boston, US, 1983) 83-87.

[133] Rosen, M. Gender differences in reading performance on documents across countries. Journal of Reading and Writing, 14, , (2001),

[134] Roto, V. Ketola, P. Huotari, S. User Experience Evaluation in Nokia. In proceedings of CHI 2008, (Florence, Italy, 2008)

[135] Roto, V. Law, e. Vermeeren, A. Hoonhout, J. User Experience White Paper: Results from Dagstuhl Seminar on Demarcating User Experience. Online supplement, Available http://www.allaboutux.org/files/UX-WhitePaper.pdf, 8.2.2011

[136] Roto, V. Rantavuo, H. Väänänen-Vaino-Mattila. K. Evaluating User Experience of Early Product Concepts. In proceedings of DPPI09, (Compiegne, France, 2009)

[137] Roto, V. Rautava, M. User Experience Elements and Brand Promise. In proceedings of NordiCHI'08 (Lund, Sweden, 2008).

[138] Roto, V. User Experience Building Blocks, In proceedings of 2nd COST294-MAUSE Workshop – Towards a Unified View (Oslo, Norway, 2006), 124-128

[139] Roto, V. Web Browsing on Mobile Phones – Characteristics of User Experience. Doctoral dissertation, Helsinki University of Technology. 2006.

[140] Royce, W. W. Managin the Development of Large Software Systems. In Proceedings of IEEE WESCON (plasex, plasey, 1970)

[141] Ryan, R. M. Deci, E. L. Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions. Contemporary Educational Psychology, 25, (2000), 54-67.

[142] Sandfeld, J. Jensen, B.R. Effect of computer mouse gain and visual demand on mouse clicking performance and muscle activation in a young and elderly group of experienced computer users. Applied Ergonomics, 36, 5, (2005), 547-555.

[143] SAP Design Guild, Golden rules for bad user interfaces, Online supplement, Available http://www.sapdesignguild.org/community/design/golden_rules.asp 13.1.2011

[144] Schach, S. R. Classical and Object-Oriented Software Engineering. McGraw-Hill, 2002. ISBN:0-0723-9559-1

[145] Schütte, S. Engineering Emotional Values in Product Design – Kansei Engineering in Development. Doctoral dissertation, Linköping University, 2005.

[146] ScienceDirect redesign: The user experience. Online supplement, Available http://www.elsevier.com/framework_librarians/WhitePapers/200606-sciencedirectredesign.pdf, 21.2.2011

[147] Shaw, M. Prospects for an Engineering Discipline of Software, IEEE Software Journal, 7, 6, (1990), 15-24

[148] Smith, A. An Inquiry into the Nature and Causes of the Wealth of Nations, 1776. Online supplement. Available http://www.econlib.org/library/Smith/smWN.html 5.9.2011

[149] Sproll, S. Peissner, M. Sturm, C. From Product Concepts to User Experience: Exloring UX Potentials at Early Product Stages. In proceedings of NordiChi 2010. (Reykjavik, Iceland, 2010)

[150] Stanford Encyclopedia of Philosophy, Cognitive Science. Online supplement. Available at http://plato.stanford.edu/archives/fall2008/entries/cognitive-science/ 6.9.2011

[151] Sun, Z. A Waterfall model for knowledge management and experience management. In proceedings of HIS'04, (Kitakyushu, Japan, 2004), 472-475

[152] Sutcliffe, A. G. Wang, I. Integrating Human Computer Interaction with Jackson System Development. The computer journal, 34, 2, (1991), 132-142

[153] Swallow, D. Blythe, M. Wright, P. Grounding Experience: Relating Theory and Method to Evaluate the User Experience of Smartphones, In proceedings of EACE'05, (Athens, Greek, 2005), 91-98

[154] Sward, D. MacArthur, G. Making User Experience a Business Strategy. In proceedigns of COST294-MAUSE, (Lancaster, UK, 2007), 35-42.

[155] Sy, D. Adapting Usability Investigation for Agile user-centered Design. Journal of Usability Studies, 2, 3, (2007), 112-132.

[156] Sykorova, D. Age indentity. Journal of Sociology, 41, (2009),

[157] Takeuchi, T. Nonaka, I. The new product development game. Harvard Business Review, January-February, (1986)

[158] Tarmo.fi blog, do not draw diagrams of wrong practices – or: Why people still believe in the Waterfall model, Online supplement, Available http://tarmo.fi/blog/2005/09/dont-draw-diagrams-of-wrong-practices-or-why-people-still-believe-in-the-waterfall-model/ 5.1.2011

[159] Tarssanen, S. Handbook for Experience Tourism Agents. University of Lapland Printing Centre, 2007. ISBN: 978-952-5585-53-0

[160] Thimbleby, H. Understanding user Centred Design (UCD) for People with Special Needs, In proceedings of ICCHP '08 (Linz, Austria, 2008), 1-17.

[161] Toyota Motor Sales. Toyota Announces Voluntary Recall on 2010 Model-Year Prius to Update ABS Software, 8.2.2010. Online supplement. Available http://www.toyota.com/recall/abs.html 11.1.2011.

[162] User Experience Network (discontinued), Online supplement. Archive available http://www.uxnet.org/ 14.2.2011

[163] Vermeeren, A. P.O.S. Law, E. L-C. Roto, V. Obrist, M. Hoonhout, J. Väänänen-Vainio-Mattila, K. User Experience Evaluation Methods: Current State and Development Needs. In proceedings of NordiChi 2010. (Reykjavik, Iceland, 2010)

[164] VIRTAHEPO – Towards participatory virtual community. Online supplement. Available http://virtual.vtt.fi/virtual/virtahepo/show/virtahepolty.pdf

[165] Vukelja, J. Müller, L. Opwis, K. Are engineers condemned to design? a survey on software engineering and UI design in Switzerland, In proceedings of INTERACT'07, (Rio De Janeiro, Brasil, 2007), 555-568

[166] Väänänen-Vainio-Mattila, K. Roto, V. Hassenzahl, M. Towards Practical User Experience Evaluation Methods. In proceedings of VUUM 2008, (Reykjavik, Iceland, 2008).

[167] Väänänen-Vainio-Mattila. K. Wäljas, M. Development of Evaluation Heuristics for Web Service User Experience. In proceedings of CHI 2009, (Boston, USA, 2009), 3679-3684

[168] WailGum, T. Stupid Users Are So Stupid, CIO Blog, Online supplement, Available http://advice.cio.com/thomas_wailgum/9923/stupid_users_are_so_stupid

[169] Webcredible, User-centered design – 6 methods. Online supplement. available http://www.webcredible.co.uk/user-friendly-resources/web-usability/user-centered-design.shtml, 4.2.2011

[170] Wilamowitz-Moellendorff, M. Hassenzahl, M. Platz, A. Dynamics of user experience: how the perceived quality of mobile phones changes over time. In proceedings of 2nd COST294-MAUSE Workshop – Towards a Unified View (Oslo, Norway, 2006), 74–78.

[171] Woodward, E. Surdek, S. Ganis, M. A Practical Guide to Distributed Scrum. IBM Press, 2010, ISBN: 0-1370-4113-6

[172] Yildirim, S. Narayanan, S. Potamianos, A. Detecting emotional state of a child in a conversational computer game. Journal of Computer Speech & Language, 25, (2011), 29-44.

[173] Zaman, B. Introducing contextual laddering to evaluate the likeability of games with children. Journal of Cognition, Technology & Work, 10, 2, (2008), 107-117.

[174] Zeller, A. Why programs fail: a guide to systematic debugging, second edition. Morgan Kaufmann, 2009, ISBN:978-12-374515-6

Appendixes

I. UX professional survey and end user survey

Background questions

Gender? *

○ Male ○ Female

Age? *

20-24

With the following scale, evaluate your ability with computers *

- ○ Very poor
- ○ I can use basic programs
- ○ I can use, install and update programs
- ○ I can develop / maintain minor programs, web sites, etc.
- ○ I can develop / maintain advanced programs, web sites, etc.
- ○ I consider myself as a rock solid professional

In your opinion, select three definitions that matches best to UX (User Experience). 1 = best match

If you don't know or don't care, jump to the bottom of this question

	1	2	3
Consequence of a user's internal state (predispositions, expectations, needs, motivation, mood, etc.), the characteristics of the designed system (complexity, purpose, usability, functionality, etc.) and the context or the environment within which the interaction occurs (organizational / social setting, meaningfulness of the activity, voluntariness of use, etc.)	○	○	○
The quality of experience a person has when interacting with a specific design	○	○	○
All aspects of the user's experience when interacting with the product, service, environment or facility	○	○	○
A special case of experience, where the person can use a system, with or without a purpose. Using means that the user not only senses the system, but also has the opportunity to manipulate or control the system	○	○	○
A momentary, primarily evaluative feeling (good-bad) while interacting with a product or service	○	○	○
The entire set of effects that is elicited by the interaction between a user and a product, including the degree to which all our senses are gratified (aesthetic experience), the meanings we attach to the product (experience of meaning), and the feelings and emotions that are elicited (emotional experience)	○	○	○
I don't know	○	○	○
I don't care	○	○	○

Think about software (application, game, etc.) and evaluate the following from the user experience point of view.

Evaluate the importance of the following UX attributes to you when using a software.

	Very Important	Important	Moderately Important	Of Little Importance	Unimportant	Don't know / don't understand
Accessibility	○	○	○	○	○	○
Aesthetics	○	○	○	○	○	○
Brand	○	○	○	○	○	○
Consistency (of functionality)	○	○	○	○	○	○
Coolness (wow effect)	○	○	○	○	○	○
Device (where software is used)	○	○	○	○	○	○
Emotions (own)	○	○	○	○	○	○
Environment (where usage happens)	○	○	○	○	○	○
Expectations (own)	○	○	○	○	○	○
Functionality	○	○	○	○	○	○
Interaction	○	○	○	○	○	○
Motivation (own)	○	○	○	○	○	○
Pleasure	○	○	○	○	○	○
Price of purchace	○	○	○	○	○	○
Reliability	○	○	○	○	○	○
Stability (no crashing, bugs, etc.)	○	○	○	○	○	○
Trust & privacy	○	○	○	○	○	○
Usability	○	○	○	○	○	○
Usefulness	○	○	○	○	○	○
User interface	○	○	○	○	○	○
Ease of taking into use	○	○	○	○	○	○

Something else you would like to say about UX, measuring it, importance, etc?

If you have any professional opinions, suggestions, etc. for the survey, questions, etc. please write those here.

II. Upper secondary survey

Note, this survey was conducted in Finnish and only the suitable parts are presented in this appendix.

Taustakysymykset:

1) Sukupuoli *

- Nainen
- Mies

2) Ikä *

16 tai alle

4) Hallitsen tietokoneiden käytön mielestäni: *

- Huonosti
- Osaan käyttää joitain ohjelmia
- Osaan asentaa, käyttää ja päivittää ohjelmia
- Osaan html:n tai jonkin ohjelmointikielen alkeet
- Olen mielestäni hyvä ohjelmoimaan
- Vastaan tasoltani ammattilaista

8) Kuinka tärkeiksi koet seuraavat käyttäjäkokemuksen ominaisuudet tai osa-alueet:

	Erittäin tärkeä	Tärkeä	Jossain määrin tärkeä	Ei juurikaan tärkeä	Ei tärkeä	En ymmärrä mitä tarkoittaa
Käytettävyys	○	○	○	○	○	○
Estetiikka	○	○	○	○	○	○
Brändi	○	○	○	○	○	○
Johdonmukaisuus	○	○	○	○	○	○
Uskottavuus kavereiden silmissä	○	○	○	○	○	○
Laite jossa ohjelmisto	○	○	○	○	○	○
Tunteet joita herättää	○	○	○	○	○	○
Odotukset	○	○	○	○	○	○
Toimivuus	○	○	○	○	○	○
Vuorovaikutus	○	○	○	○	○	○
Motivaatio	○	○	○	○	○	○
Hinta	○	○	○	○	○	○
Luotettavuus	○	○	○	○	○	○
Vakaus	○	○	○	○	○	○
Turvallisuus, tietoturva	○	○	○	○	○	○
Merkki	○	○	○	○	○	○
Hyödyllisyys	○	○	○	○	○	○
Käyttöliittymä	○	○	○	○	○	○
Käyttöönoton helppous	○	○	○	○	○	○
Pitkäikäisyys	○	○	○	○	○	○
Ihkuus	○	○	○	○	○	○
Opittavuus	○	○	○	○	○	○
Tehokkuus	○	○	○	○	○	○
Muistettavuus	○	○	○	○	○	○
Käytön virheiden minimointi	○	○	○	○	○	○
Tyytyväisyys	○	○	○	○	○	○
Visuaalisuus	○	○	○	○	○	○

III. .NET Code Camp survey

By answering to this survey you can help me to further enhance the Wizard of UX, so it will better serve for its purpose. Given responses might also be used in my dissertation work, anonymously of course.

Anssi Jääskeläinen
LUT/IT

Background questions

Age? *

- ○ Under 20
- ○ 20-29
- ○ 30-39
- ○ 40-49
- ○ 50-59
- ○ 60 or over

Gender? *

○ Male ○ Female

Evaluate the following statements by using the scale below

CA = Completely Agree
MA = Moderately Agree
SA = Slightly Agree
N = Neither
SD = Slightly Disagree
MD = Moderately Disagree
CD = Completely Disagree
DN = Don't know

	CA	MA	SA	N	SD	MD	CD	DN
I could increase my know-how about user experience by using the Wizard of UX?	○	○	○	○	○	○	○	○
I could raise my understanding about end user(s) viewpoint by using the Wizard of UX?	○	○	○	○	○	○	○	○
End user demographics have impact to UX?	○	○	○	○	○	○	○	○
Some baseline(s) for my development project could be founded by using the Wizard of UX?	○	○	○	○	○	○	○	○
The Wizard of UX could aid me in my work?	○	○	○	○	○	○	○	○
Current functionality of the Wizard of UX fills my needs?	○	○	○	○	○	○	○	○
The Wizard of UX could be a useful tool for me?	○	○	○	○	○	○	○	○
The Wizard of UX could be a useful tool for my company?	○	○	○	○	○	○	○	○
I would recommend Wizard of UX to my UX friends or my UX oriented co-workers?	○	○	○	○	○	○	○	○

ACTA UNIVERSITATIS LAPPEENRANTAENSIS

417. SAMPO, JOUNI. On convergence of transforms based on parabolic scaling. 2010. Diss.

418. TURKU. IRINA. Adsorptive removal of harmful organic compounds from aqueous solutions. 2010. Diss.

419. TOURUNEN, ANTTI. A study of combustion phenomena in circulating fluidized beds by developing and applying experimental and modeling methods for laboratory-scale reactors. 2010. Diss.

420. CHIPOFYA, VICTOR. Training system for conceptual design and evaluation for wastewater treatment. 2010. Diss.

421. KORTELAINEN, SAMULI. Analysis of the sources of sustained competitive advantage: System dynamic approach. 2011. Diss.

422. KALJUNEN, LEENA. Johtamisopit kuntaorganisaatiossa – diskursiivinen tutkimus sosiaali- ja terveystoimesta 1980-luvulta 2000-luvulle. 2011. Diss.

423. PEKKARINEN, SATU. Innovations of ageing and societal transition. Dynamics of change of the socio-technical regime of ageing. 2011. Diss.

424. JUNTTILA, VIRPI. Automated, adapted methods for forest inventory. 2011. Diss.

425. VIRTA, MAARIT. Knowledge sharing between generations in an organization – Retention of the old or building the new 2011. Diss.

426. KUITTINEN, HANNA. Analysis on firm innovation boundaries. 2011. Diss.

427. AHONEN, TERO. Monitoring of centrifugal pump operation by a frequency converter. 2011. Diss.

428. MARKELOV, DENIS. Dynamical and structural properties of dendrimer macromolecules. 2011. Diss.

429. HÄMÄLÄINEN, SANNA. The effect of institutional settings on accounting conservatism – empirical evidence from the Nordic countries and the transitional economies of Europe. 2011. Diss.

430. ALAOUTINEN, SATU. Enabling constructive alignment in programming instruction. 2011. Diss.

431. ÅMAN, RAFAEL. Methods and models for accelerating dynamic simulation of fluid power circuits. 2011. Diss.

432. IMMONEN, MIKA. Public-private partnerships: managing organizational change for acquiring value creative capabilities. 2011. Diss.

433. EDELMANN, JAN. Experiences in using a structured method in finding and defining new innovations: the strategic options approach. 2011. Diss.

434. KAH, PAUL. Usability of laser - arc hybrid welding processes in industrial applications. 2011. Diss.

435. OLANDER, HEIDI. Formal and informal mechanisms for knowledge protection and sharing. 2011. Diss.

436. MINAV, TATIANA. Electric drive based control and electric energy regeneration in a hydraulic system. 2011. Diss.

437. REPO, EVELIINA. EDTA- and DTPA-functionalized silica gel and chitosan adsorbents for the removal of heavy metals from aqueous solutions. 2011. Diss.

438. PODMETINA, DARIA. Innovation and internationalization in Russian companies: challenges and opportunities of open innovation and cooperation. 2011. Diss.

439. SAVITSKAYA, IRINA. Environmental influences on the adoption of open innovation: analysis of structural, institutional and cultural impacts. 2011. Diss.

440. BALANDIN, SERGEY, KOUCHERYAVY, YEVGENI, JÄPPINEN, PEKKA, eds. Selected Papers from FRUCT 8 .2011.

441. LAHTI, MATTI. Atomic level phenomena on transition metal surfaces. 2011. Diss.

442. PAKARINEN, JOUNI. Recovery and refining of manganese as by-product from hydrometallurgical processes. 2011. Diss.

443. KASURINEN, JUSSI. Software test process development. 2011. Diss.

444. PEKKANEN, PETRA. Delay reduction in courts of justice – possibilities and challenges of process improvement in professional public organizations. 2011. Diss.

445. VANHALA, MIKA. Impersonal trust within the organization: what, how, and why? 2011. Diss.

446. HYNYNEN, KATJA. Broadband excitation in the system identification of active magnetic bearing rotor systems. 2011. Diss.

447. SOLONEN, ANTTI. Bayesian methods for estimation, optimization and experimental design. 2011. Diss.

448. JABLONSKA, MATYLDA. From fluid dynamics to human psychology. What drives financial markets towards extreme events. 2011. Diss.

449. MYÖHÄNEN, KARI. Modelling of combustion and sorbent reactions in three-dimensional flow environment of a circulating fluidized bed furnace. 2011. Diss.

450. LAATIKAINEN, MARKKU. Modeling of electrolyte sorption – from phase equilibria to dynamic separation systems. 2011. Diss.

451. MIELONEN, JUHA. Making Sense of Shared Leadership. A case study of leadership processes and practices without formal leadership structure in the team context. 2011. Diss.

452. PHAM, ANH TUAN. Sewage sludge electro-dewatering. 2011. Diss.

453. HENNALA, LEA. Kuulla vai kuunnella – käyttäjää osallistavan palveluinnovoinnin lähestymistavan haasteet julkisella sektorilla. 2011. Diss.

454. HEINIMÖ, JUSSI. Developing markets of energy biomass – local and global perspectives. 2011. Diss.

455. HUJALA, MAIJA. Structural dynamics in global pulp and paper industry. 2011. Diss.

456. KARVONEN, MATTI. Convergence in industry evolution. 2011. Diss.

457. KINNUNEN, TEEMU .Bag-of-features approach to unsupervised visual object categorisation. 2011. Diss.

458. RUUSKANEN, VESA. Design aspects of megawatt-range direct-driven permanent magnet wind generators. 2011. Diss.

459. WINTER, SUSANNA. Network effects: scale development and implications for new product performance. 2011. Diss.