Thesis for The Degree of Doctor of Philosophy

Integrating User eXperience Principles and Practices into Software Development Organizations:
An Empirical Investigation

Pariya Hajar Kashfi

Division of Software Engineering
Department of Computer Science & Engineering
Chalmers University of Technology and Gothenburg University
Gothenburg, Sweden, 2018

Integrating User eXperience Principles and Practices into Software Development Organizations: An Empirical Investigation

Pariya Hajar Kashfi

ISBN 978-91-7597-833-8
Doktorsavhandlingar vid Chalmers tekniska högskola. Ny serie nr 4514
ISSN 0346-718X

Technical Report No 166D
Department of Computer Science and Engineering
Division of Software Engineering
Chalmers University of Technology and Gothenburg University
Gothenburg, Sweden

This thesis has been prepared using LaTeX.
Printed by Chalmers Reproservice,
Gothenburg, Sweden 2018.

To Mohsen,
my parents & my brothers

Abstract

Background: To be effective, User eXperience (UX) principles and practices need to be integrated into development processes and organizations, what we refer to as *UX integration*. However, software companies often face various challenges that hinder a successful UX integration.

Objective: The aim of this thesis is to facilitate and improve the current state of UX integration in the software industry. To that end, we present an empirical investigation of current UX integration challenges and success factors and analyze them in relation to other software quality characteristics, in particular, usability.

Method: We performed a series of studies, mainly in the Swedish software industry and applied a variety of methods including interviews, observations, and workshops. We used Grounded Theory (GT) and thematic analysis to drive our data gathering and to analyze our data.

Results: We showed that UX integration challenges and success factors are both technical and organizational, however, they mainly belong to the latter category. We found that various decisions that are made outside the authority of UX practitioners have an inevitable impact on enabling or prohibiting UX integration and that the integration is influenced by various changes that organizations undergo over time as well as planned UX initiatives. Our findings underline the similarities between UX integration and organizational change, in general, and Software Process Improvement (SPI) in particular. We also found that the known unique characteristics of UX (*subjective*, *holistic*, *dynamic*, *context-dependent*, and *worthwhile*) have implications not only for the day-to-day work of practitioners but also for UX integration. Based on our findings, we propose various *UX integration principles and practices* to help software companies in their integration efforts.

Conclusion: We argue that to prevent a lopsided focus on the pragmatic aspect of UX in the software industry, software practitioners and researchers should explicitly differentiate between UX and other software quality characteristics, in particular, usability and address the unique characteristics of UX in their work. In addition, they should apply the existing body of knowledge in the two fields of organizational change and SPI especially to address the organizational issues concerning UX integration. Although our focus has been on UX, our findings also may shed light on integrating other multi-disciplinary and emerging concepts into the complex context of software organizations.

Keywords

User eXperience, Usability, Quality Characteristics, Software Process Improvement, Organizational change, Empirical Research

Acknowledgment

I would like to express my gratitude to all of the individuals whose support and help has made this research a reality; especially my supervisor, Robert Feldt, and my co-supervisor, Agneta Nilsson. My gratitude also extends to Olof Torgersson, my previous supervisor, who gave me a chance to pursue research. I also convey my appreciation to all the practitioners who participated in my studies.

I would also like to convey my gratitude to my colleagues, including all the administrative staff, in the Division of Software Engineering, in particular, Emil Alégroth, Hiva Allahyari, Lucas Gren, Erik Knauss, Grischa Liebel, Antonio Martini, Richard Berntsson Svensson. I also want to thank my friends and colleagues Sus Lyckvi, Cecilia Berlin, and Ana Magazinius who provided valuable feedback on my research.

My dear friends, in particular, Maryam, Minou, Mohsen, Sayeh, and Shiva, know that they all hold a very special place in my heart and I am grateful to them for all of the joy they have brought into my life over the past few years. But my supportive family–my parents and brothers–know that they are and always will be a part of my heart. I hope that the realization that their love, patience, and willingness to bear the burden of our separation, has empowered me to make my dreams come true will allow them to look upon my achievements here with pride.

And finally, thanks to you, my dearest *Mohsen*. You are not only a marvelous life partner but also a tremendous friend. I appreciate the fruitful discussions we have had together and the feedback you provided on this thesis. I am grateful to you for being always cheerful and supportive through both sunshine and rain, as well as all of the other moments that will forever remain between you and I.

List of Publications

Appended papers

This thesis is primarily supported by the following papers:

1. P. Kashfi, A. Nilsson, R. Feldt, “Integrating User eXperience Practices Into Software Development Processes: Implications of The UX Characteristics"
   PeerJ Computer Science (an Open Access Journal), 3, e130, 2017.

2. P. Kashfi, A. Nilsson, R. Feldt, R. Berntsson Svensson, “A Conceptual UX-aware Model of Requirements"
   6th International Working Conference on Human-Centred Software Engineering, 234-245, 2016.

3. P. Kashfi, R. Feldt, A. Nilsson, “Integrating UX Principles and Practices into Software Development Organizations: A Case Study of Influencing Events”
   In submission to The Journal of Systems and Software.

4. P. Kashfi, R. Feldt, A. Nilsson, “Integrating UX Principles and Practices into Software Development Organizations: A Case Study of Challenges and Success Factors"
   In submission to The Journal of Systems and Software.

5. P. Kashfi, R. Feldt, A. Nilsson, R. Berntsson Svensson, “Evidence-based Timelines for User eXperience Software Process Improvement Retrospectives: A Case Study of User eXperience Integration"
   42nd Euromicro Conference on Software Engineering and Advanced Applications, 59-62, 2016.

6. P. Kashfi, K. Kuusinen, R. Feldt, “Stakeholder Involvement: A Success Factor for Achieving Better UX Integration"
   1st Workshop on Stakeholder Involvement in Agile Development, NordiCHI, 2016

Other papers

The following papers are published but not appended to this thesis, either due to overlapping contents to the appended papers, contents not related to the thesis or because the contents are of less priority for the thesis main conclusions.

1. P. Kashfi, A. Nilsson, R. Feldt, “Supporting practitioners in prioritizing user experience requirements"
   3rd International Workshop on Requirements Prioritization for Customer Oriented Software Development, 23, 2012.

2. H. Kashfi, “The Intersection of Clinical Decision Support and Electronic Health Record: A Literature Review"
   1st International Workshop on Interoperable Healthcare Systems, 347-353, 2011.

3. H. Kashfi, O Torgersson, “Supporting openEHR Java Desktop Application Developers"
   Medical Informatics in a United and Healthy Europe, 724-728, 2011.

4. H. Kashfi, J Robledo Jr., “Towards a Case-Based Reasoning Method for openEHR-Based Clinical Decision Support"
   3rd International Workshop on Knowledge Representation for Health Care, 2011.

5. H. Kashfi, “Applying a user centered design methodology in a clinical context”
   13th International Congress on Medical Informatics, 927-931, 2010.

6. H. Kashfi, O. Torgersson, “A Migration to an openEHR-Based Clinical Application”
   Medical Informatics in a United and Healthy Europe, 152-156, 2009.

7. H. Kashfi, “An openEHR-Based Clinical Decision Support System: A Case Study”
   Medical Informatics in a United and Healthy Europe, 150, 2009.

Statement of contribution

In all listed papers, the first author was the primary contributor to the research idea, design, data collection, analysis and/or reporting of the research work.

Contents

Abstract **v**

Acknowledgement **vii**

List of Publications **ix**

1 Introduction **1**

1 Introduction 1

2 Background 5

2.1 Software Quality Models and Their Relation to The Concept of UX 5

2.2 Usability 6

2.3 User eXperience (UX) 6

2.4 Hassenzahl's Model of UX and UX Characteristics . . . 8

2.5 From User-Centered Design (UCD) to Experience-Centered Design 9

3 Related Work 13

4 Problem Statement and Main Objectives 16

5 Research Design 17

6 Overview of The Appended Papers 22

6.1 Paper I 22

6.2 Paper II 24

6.3 Paper III 25

6.4 Paper IV 26

6.5 Paper V 28

6.6 Paper VI 28

7 Discussion 29

8 Threats to Validity 32

9 Conclusions and Future Work 34

Bibliography **37**

2 Paper I **45**

1 Introduction 46

2 Background and Related Work 47

3 Methods 51

3.1 Research Sites 51

3.2 Data Collection 52

3.3 Data Analysis 53
3.4 Threats to Validity 53
4 Results 55
5 Discussion 72
6 Conclusion 78

3 Paper II 87
1 Introduction 88
2 Research Approach 89
3 Results and Analysis 90
4 Discussion 93

4 Paper III 101
1 Introduction 102
2 Background 103
3 Research Method 105
3.1 The Case Company 105
3.2 Limitations and Threats to Validity 109
4 Results 111
4.1 Period One (1993-2002): Focusing on Graphical User Interfaces 111
4.2 Period Two (2003-2006): Initiating Usability Integration 113
4.3 Period Three (2007-2012): Initiating UX Integration . . 116
4.4 Period Four (2013-2015): Improving UX Integration . . 119
5 Discussion 124
5.1 Lessons Learned Concerning UX Integration 125
5.2 Pitfalls in Transition from Usability to UX Integration . 127
6 Conclusion 131

5 Paper IV 139
1 Introduction 140
2 Background and Related work 141
3 Research Method 144
3.1 The Case Company 144
3.2 Data Gathering 146
3.3 Data Analysis 147
3.4 Limitations and Threats to Validity 148
4 Results 150
4.1 Theme One: Beliefs and Attitudes Concerning Software Quality and UX 150
4.1.1 Challenges and success factors concerning the value of software quality in general and UX in particular 150
4.1.2 Challenges and success factors concerning the scope of UX 151
4.1.3 Challenges and success factors concerning the importance of hedonic versus pragmatic aspect of UX 152

4.1.4 Challenges and success factors concerning the evaluation of UX 152
4.2 Theme Two: Business Alignment of UX Integration . . 154
4.2.1 Challenges and success factors concerning the relation of UX to the business strategy 154
4.2.2 Challenges and success factors concerning the importance of the needs of users versus customers 155
4.3 Theme Three: UX practices and Responsibilities 158
4.3.1 Challenges and success factors concerning UX leadership and authority 162
4.3.2 Challenges and success factors concerning politics of UX integration 165
5 Discussion 167
6 Conclusion and Future Work 175

6 Paper V 181
1 Introduction 182
2 Methods 183
3 Results and Analysis 184
4 Discussion 186

7 Paper VI 191
1 Introduction 192
2 Background 192
3 Method 193
4 Results 194
4.1 A Guideline on Practicing UX 194
4.1.1 Dynamic Nature of UX and its Practical Implications 194
4.1.2 Subjective Nature of UX and its Practical Implications 196
4.1.3 Holistic Nature of UX and its Practical Implications 197
4.1.4 Context-dependent Nature of UX and its Practical Implications 197
4.2 A Guideline on UX Integration 198
5 Discussion 201
6 Conclusion 202

Chapter 1

Introduction

1 Introduction

It is known that delivering many functions is not always enough for the business success of interactive software [1, 2]. Often, to deliver value to various stakeholders, a large number of software quality characteristics need to be considered in addition to various functions [2, 3]. A variety of such necessary and desired software quality characteristics are defined and categorized in ISO/IEC 25010 (system and software quality models) [2]. These characteristics include those related to the software system and data (internal and external quality also known as product quality) as well as the impact the system has on its stakeholders including the user (quality in use) [2].

ISO/IEC 25010 [2] describes that *"the quality in use (QiU) of a system characterizes the impact that the product (system or software product) has on stakeholders. It is determined by the quality of the software, hardware and operating environment, and the characteristics of the users, tasks and social environment. All these factors contribute to the quality in use of the system."* The concept of QiU resembles the concept of User eXperience (UX) in our view. However, compared to the QiU model, the existing UX models entail a broader view on various aspects of software use and, in particular, the important role of the user and her psychological state and needs on the 'impact' the software may have. These models also to a greater extent differentiate the notions of satisfaction, pleasure, appeal, and emotional consequences of software use [4].

UX is defined by ISO/IEC 9241 [5] as *"person's perceptions and responses resulting from the use and/or anticipated use of a product, system or service."*. This ISO/IEC further emphasizes that *"user experience is a consequence of the presentation, functionality, system performance, interactive behavior, and assistive capabilities of an interactive system, both hardware and software. It is also a consequence of the user's prior experiences, attitudes, skills, habits, and personality."* Another well-known definition of UX is the one by Hassenzahl and Tractinsky [4]: *"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.)."*

Both these definitions share the same essence and emphasize the important role of the user, her internal state, and both functionality and quality characteristics of a piece of software in forming an experience as a consequence and impact of use.

Admittedly, the term UX is associated with a wide variety of meanings [6] which range from traditional usability to beauty, hedonic, affective, or experiential aspects of technology use [4]. Empirical research shows that the perception of UX is generally different in academic and industrial contexts: whereas the former concentrates on hedonism and emotions, the latter focuses more on functional and usability issues [7]. Most importantly, the term UX is used not only as a software quality characteristic but also to refer to the phenomenon of experience in the context of technology usage [8].

While acknowledging the different usages of the term UX, in this thesis, we use this term to refer to the software quality characteristic that relates to the impact the software has on its users and the consequences of this impact. Hence, similarly to describing a software with more traditional quality characteristics, e.g. this software has high performance and security, one may describe a software as 'this software delivers a good UX' which, in other words, means the consequences of the software use are positive from the perspective of the users.

The reason why we emphasize using the term UX as a quality characteristics is to make the abstract phenomenon of experience more tangible for the software development practitioners and enable comparing this concept with other more known quality characteristics, in particular, with regards to their industrial practice. However, in contrary to other quality characteristics, in particular, usability, UX only comes about through the use of (interactive) software [8]. Therefore, similarly to QiU, UX can never be formed, observed, or measured without the presence of a user when putting the software to use in a certain context. In addition, practitioners can manipulate UX mainly by manipulating functions or a number of other quality characteristics including, for instance, usability or performance [2, 8, 9]. This manipulation can never guarantee a certain UX but only increase the likelihood of delivering a good UX when it is based on, among others, a deep understanding of the end users', their preferences, goals, values, and the context in which the software is going to be used [8, 9].

Usability is often seen as a necessary precondition for good UX yet different from it [10,11]. Similarly, in ISO/IEC 25010, usability is categorized as one of the external quality characteristics that influence QiU. One of the widely used definitions of usability is given by ISO/IEC 9241 [5]: *"the extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use."* UX has five unique characteristics that differentiate it from usability (and all quality characteristics for that matter): *subjective* (heavily relying on human perception), *holistic* (including both hedonic and pragmatic aspects of use), *dynamic* (changing over time), *context-dependent* (situated in context) and *worthwhile* (encompassing positive and meaningful consequences of use) [8].

Although practitioners cannot guarantee a specific experience, applying certain principles and practices can increase the likelihood of delivering a good UX [8]. We refer to such principles and practices as *UX principles and prac-*

tices. Here, by *principle* we mean “a comprehensive and fundamental law, doctrine, or assumption” [12]. Principles provide the basis for many different software practices [13] and are important factors and fundamental concepts that practitioners need to take into account in their work. UX principles, in fact, reflect the understanding of UX as a phenomenon. Examples are: both hedonic and pragmatic aspect of software use play an important role in forming UX, UX is temporal, etc.

We separate *principles* from *practices*, activities that practitioners need to perform in order to satisfy the principles [13]. Practices are performed throughout the life-cycle of a software system and in different steps of the process (analysis, design, development, evaluation). Examples of UX practices are: identify users’ personal goals and preferences, create prototypes, involve users in the design process, evaluate the software from both pragmatic and hedonic perspectives, etc.

Applying UX principles and practices in isolation is not enough and, as empirical research findings show, early and continuous attention to them is required to ensure delivering a good UX through the developed software [14–16]. Hence, UX principles and practices need to be integrated into the development processes and considered early on and throughout projects in order to have an impact [17, 18]. We refer to *the timely process of integrating UX principles and practices into development processes and organizations* as *UX integration*. Here, by integration, we emphasize making these principles and practitioners an integral part of the development processes and not merely add-ons. UX principles and practices should be adjusted to and aligned with already existing software development principles and practices. Most importantly, it is not enough to introduce them only in later stages of software development, rather, organizations need an early and continuous commitment to these principles and practices for them to have an impact [17, 18].

The concept of integration is not exclusive to UX principles and practices and has been previously argued for in various sub-domains of software development and engineering. In the field of requirements engineering, for instance, Nusibeh and Easterbrook [19] highlight the importance of integrating different requirements activities (i.e., eliciting, modeling and analyzing, communicating, agreeing, and evolving requirements) into a single development process in order to maximize their impact and enable their effective management. In Nusibeh and Easterbrook’s view, such an integration is, in particular, important since in performing requirements activities, practitioners need to apply various methods and take different viewpoints into account.

Similarly, researchers emphasize that practices to support quality requirements should be integrated into development processes early and continuously [20, 21]. In particular, since functional and quality requirements constrain each other and are realized through architectural decisions, it is recommended to treat them together and in a tightly integrated and coherent approach [21]. Studies show that in lack of an integration, software companies may fail to consider the trade-offs between functionality and quality characteristics; they may also fail to find a right balance among competing software quality characteristics. The companies may also dismiss these characteristics in later phases, or fail to plan and rely on them to achieve competitive advantages [20, 22]. For instance, Berntsson Svensson et al. [20] found that since requirements on these quality characteristics are not taken into consideration during product planning, they are thus not included as hard requirements (requirements that must be met) in the projects. They also found that close to 1 out of 5 requirements on these quality characteristics are dismissed from the projects at some stage during development.

In yet another area, researchers in software security have proposed a number of approaches to integrate security engineering practices into software development processes, in particular, agile methodologies like Scrum and XP [23–25]. These proposals build on the common premise that integration is required in order to ensure that security requirements get due attention. Usability is another quality characteristic for which researchers have emphasized the importance of integration [26–28]. Research shows that to achieve a high usability in software, practitioners require to take usability into account from early stages of projects [26, 27] and address the organizational as well as technical issues concerning integration [29].

A number of studies have investigated UX integration and reported on challenges and success factors that can prohibit or enable a successful UX integration [17, 30–32]. Like usability integration, UX integration is known to be a socio-technical endeavor [18, 33] which requires organizational as well as technical changes, e.g. introducing new roles, modifying development processes or introducing new tools and methods.

Despite the importance of UX and UX integration, many companies still face various challenges that prevent them from achieving a sustainable UX integration [7, 34, 35], an integration successful not only in a short period of time but also maintained over time. A better understanding of UX integration challenges and success factors can help systematically addressing them to facilitate and improve the current state of UX integration in the software industry. As we mentioned before, the topic of integration has been previously studied also for usability and other software quality characteristics. Therefore, we can gain a better understanding of UX integration by analyzing its challenges and success factors in relation to those previously reported for usability or other quality characteristics. Our understanding can also be improved by analyzing how organizations move from only developing user interfaces to also paying attention to usability and more recently UX. The insights that can be gained from such analyses could help the software community to utilize the existing knowledge on integration in other contexts also in the context of UX integration. Nevertheless, such analyses of UX integration challenges and success factors are rare in existing research.

Therefore, we performed this thesis work to address the following research question:

> *How can software companies integrate UX principles and practices into their development processes and organizations?*

This thesis includes a collection of empirical studies reported in six papers. **Paper I** contains an explorative interview study through which we identified a variety of challenges practitioners face in relation to UX integration. **Paper II** includes a model of requirements that facilitates addressing a number of challenges identified in Paper I, including communication and collaboration between UX and non-UX practitioners[1]. To further investigate the identified challenges, we performed a case study in a Swedish company. The results of this case study are presented in Paper III, Paper IV, and Paper V. **Paper III** focuses on the timeline of internal and external events that over the years enabled or prohibited UX integration in the case company. **Paper IV** discusses various identified challenges and success factors and their impact on UX integration in the case company. **Paper V** reports on a retrospective method that we proposed and applied in the case company for reflecting on UX integration efforts. **Paper VI** focuses on the involvement of stakeholders in UX integration in agile settings.

This introductory chapter provides a background to the papers and describes the relationships between them.

2 Background

To deliver a system that is consistent and of high quality, practitioners often need to take not only functions but also a large number of quality characteristics into account in development [2, 36]. Some of the software quality characteristics are internal or relate to the development process and mainly concern developers (e.g. traceability) while others are critical for the end users (e.g. performance, UX, and usability) [2]. We elaborate on the concepts of usability and UX in the following sections. This thesis targets both Software Engineering (SE) and Human-Computer Interaction (HCI) communities, therefore, it is important to understand how the concept of UX has been defined and standardized in these two fields and in relation to the main existing software quality models.

2.1 Software Quality Models and Their Relation to The Concept of UX

In SE, the concept of Quality in Use (QiU) is the closest to the concept of UX (for the definitions of these terms please see Section 1 Introduction). QiU is a model of quality characteristics that concerns how a product or system is used by the end users to satisfy their needs to achieve specific goals. QiU concerns five characteristics of software that relate to outcomes of interaction with a system: effectiveness, efficiency, satisfaction, freedom from risk, and context coverage. QiU is similar to UX in that it also emphasizes users' personal (aka. non-task-related) needs and emotional reactions. This model includes 'pleasure' (i.e., an emotional consequence of interacting with a piece of software) as a quality characteristic and defines it as: *"degree to which a user obtains pleasure from fulfilling their personal needs"*. Admittedly, practitioners need to also take into account the static properties of software and dynamic properties of the computer system which inevitably influence the experience of users. ISO/IEC 25010 groups such properties as *Product Quality* (PQ) model which includes functional suitability, performance efficiency, compatibility, **usability**, reliability, security, maintainability, and portability. In our view, the

[1] We use the terms 'UX practitioner' and 'non-UX practitioners' to respectively refer to practitioners who have UX-related roles and responsibilities and those who do not.

concept of QiU shares the same viewpoint as the concept of UX, however, it does not to a complete extent include or reflect the essence of UX. Therefore, a more comprehensive understanding of the concept of UX requires studying UX models and frameworks presented in the field of HCI (see Section 2.3).

2.2 Usability

The concepts of usability and UX have their roots in the field of Human-Computer Interaction (HCI), the field *"concerned with the design, evaluation, and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them"* [37]. A great deal of effort in this field traditionally aimed at designing and developing more usable computer systems. The field of SE has also long embraced the concept of usability as one of software quality characteristics. As we mentioned before, in ISO/IEC 25010, usability is one of the quality characteristics included in the PQ model.

Usability is known to be an important factor in designing interactive systems (for the definition please see Section 1 Introduction). Higher usability of software systems leads to increased productivity, reduced errors, and less need for training and support [38]. If the system is not usable enough for the intended users, it is likely that they do not use the system so often (underuse) or use the system improperly (misuse) and stick to their current methods for accomplishing the tasks, that both can bring costs to the organization or ruin the reputation of the team or company that developed the system [38].

In the above definition, 'user satisfaction' refers to freedom from discomfort and positive attitudes towards the use of the product. Usability has been traditionally associated with ease of use which according to ISO/IEC 9241 [5] is a 'misconception'. Although this ISO/IEC differentiates the two concepts of usability and UX and provides different definitions for them, it still emphasizes that if usability is interpreted from the perspective of the users' personal goals, as with UX, it can include perceptual and emotional aspects of use.

In this thesis, however, we use the term usability to refer to its most commonly used meaning: the level of effectiveness, efficiency, and *satisfaction* of users in performing tasks. The term UX, on the contrary, is used to also cover aspects related to *pleasure*, users' personal goals, psychological needs, and emotional reactions as elaborated below. Both UX and usability belong to the group of software quality characteristics.

2.3 User eXperience (UX)

Current research highlights that the users' overall judgment of software (related to QiU) is not merely influenced by usability rather also by how users perceive satisfaction of their personal needs such as 'being stimulated', 'gaining pleasure', or 'feeling connected to their loved ones' [9]. Therefore, to improve our understanding of users' perception of products and services, researchers have introduced the concept of UX [1,39] (for the definition please see Section 1 Introduction).

Researchers in business and economy argue that consumers expect and desire 'experiences' as well as services and goods [40]. Hence, to be profitable, businesses should explicitly design and promote ‘experiences’ [40]. Delivering a good experience not only improves people’s lives but also the viability of the companies [41]. It, therefore, is one main reason why some products outsell their competitors, even those competitors that deliver more functions (e.g. iPhone vs. Blackberry) [42]. Similarly, software companies have to deliver a unique experience to users of their software in order to survive in the competitive market [43].

The software industry has been increasingly recognizing the business value of UX [41,43]. Companies such as Google and Apple are good examples of how introducing excellent UX contributes to huge business success, for instance by attracting consumers and increasing their loyalty [8,44]. Software companies are increasingly motivated to improve the UX delivered through their developed software [43]. Early and continuous focus on UX is also known to be a success factor for start-ups [45] where UX needs to be taken into account throughout the life-cycle of software, along with other business concerns [46].

Although experience is unique, it is at the same time emerging from distinct elements and processes that can be manipulated by designers [8]. Such an experience is, as emphasized by Hassenzahl, unique to the situation- time and context: *“Experience emerges from the integration of action, perception, motivation, and emotion, however, all being in a dialog with the world at a particular place and time.”*

Hassenzahl and Tractinsky [4] categorized existing approaches to modeling UX into three groups: *the experiential*, *beyond the instrumental*, and *emotion and affect.* These approaches mainly differ in their view on how various underlying elements and processes contribute to forming the end user’s experience. In addition to the above groups, there are models that have an integrated approach to UX [47]. These different groups of approaches are described below:

- *The experiential*: Models in this approach (e.g. [6,48]) emphasize the *situatedness* and *temporality* of UX, and most importantly its ‘holistic nature’. These models emphasize that experience is not dividable into elements and is a unique combination of complexly interrelated, inseparable factors, e.g., users’ expectations and properties of the product.

- *Beyond the task-related aspect of software use (aka. beyond the instrumental)*: Models in this approach (e.g. [1]) emphasize breaking down UX into a number of underlying elements. They argue that although UX is not fully predictable, it is to some extent *shapeable* through the control of these underlying elements. In this approach, a positive UX is argued to be facilitated through *satisficing*[2] the end users’ non-task-related (aka. non-instrumental or hedonic) needs, as well as their task-related (aka. instrumental or pragmatic) needs.

- *Emotion, mood, and affect*: Models in this approach (e.g. [39,49]) emphasize human emotions and aim to understand the role of affect which not only is an antecedent to experience, but also a consequence of it. In this approach, a positive UX is argued to be facilitated through controlling and evoking positive emotions in users. These approaches often

[2] as opposed to satisfying functional requirements [3]

concentrate on specific emotions (such as pleasure, fun, or flow) as design goals.

- *Integrated experience*: Models in this approach (e.g. [50, 51]) combine the two latter approaches, through integrating task-related and non-task-related aspects of experience, as well as emotional user reactions, to achieve an integrated user experience perspective. One example is the work of Thüring and Mahlke's [51] in which UX is divided into three main components of *instrumental* (concerning usability and usefulness), *non-instrumental* (concerning look and feel) and *emotional reactions*. They emphasize that perception of users from the instrumental and non-instrumental qualities of a piece of software can lead to *episodes of subjective feelings* (i.e., emotions). These repeatedly occurring episodes at the end shape the user's *emotional experience*.

Among these current models of UX, in our view, the model presented by Hassenzahl [1] is one of the most comprehensive models. Hassenzahl differentiates the hedonic and pragmatic aspects of UX and also defines a number of unique characteristics that separate and clearly differentiate the two concepts of UX and usability. In our analysis of UX integration challenges and success factors, we have used Hassenzahl's model of UX which we describe below.

2.4 Hassenzahl's Model of UX and UX Characteristics

Hassenzahl [1] includes three key elements in his model of UX, namely 'product feature', 'product character', and 'emotional consequences'. *Product character* is a high-level description and summarizes a product attribute, e.g. novel, useful, predictable. *Product feature* is what a designer chooses in terms of content, presentation style, functionality, and interaction style in order to 'realize' the intended product character. In this model, product character itself is divided into hedonic (non-task-related) and pragmatic (task-related) *product attributes*. Pragmatic product attributes enable users to manipulate the environment. This is, for instance, achieved through functions and how these functions are implemented (i.e. their qualities such as usability, performance). Hedonic product attributes, on the contrary, relate to users' individual psychological well-being and are divided into three categories of stimulation (relates to personal development), identification (relates to self-expression and identity), and evocation (relates to memory).

In Hassenzahl's view, a designer may aim to achieve certain product characters, both hedonic and pragmatic (e.g. the product should be fun and useful), through picking a number of product features to 'fabricate' those characters (e.g. bright colors should be used in the GUI design). When a user is confronted with a product in a specific situation (time and context) she then forms a judgment of the product features (function, content, interaction, and presentation) that will form a number of emotional consequences (e.g. satisfaction, pleasure, or appeal) or behavioral consequences (e.g. spending more time with the product, or recommending it to others).

Hassenzahl's model separates satisfaction, pleasure, and appeal which all are emotional consequences of use. These emotional consequences are either 'expectation-based' (i.e. satisfaction) or 'positive well-being based' (i.e. pleasure). In Hassenzahl's view, satisfaction is linked to achieving 'specific behavioral goals' while joy or pleasure require no expectations and are achieved when something 'desirable but unexpected' is encountered. According to Hassenzahl [1]: *"if people hold expectations about the outcome of using a particular product and these expectations are confirmed they will feel satisfied."* If a product is satisfying or pleasurable then it is appealing.

Hassenzahl highlights unique characteristics of UX as being subjective, holistic, dynamic, context-dependent, and worthwhile. UX is dominated by **subjective** aspects of human perception [8, 9]. For instance, one user may perceive particular software features as simple, novel, and admirable while another may perceive them as complicated and old. UX is **holistic** which means delivering a good UX relies on satisfying both to-do and to-be goals [8]. The holistic nature of UX also means that UX is not totally reducible to its complexly intertwined underlying elements [8]. Hence, practitioners can only to some extent increase the likelihood of delivering a certain UX through manipulating its underlying elements [4, 8, 9]. UX is **dynamic** and emerges and changes over time [8]. For example, over time, the user may find a novel feature as old, or a complex feature as simple. Hence, in designing and evaluating UX, practitioners should pay certain attention to different episodes of experience [8]; namely expected experience (before usage), momentary experience (during usage), remembered experience (shortly after usage) and accumulated experience (over a longer period of use) [8]. Practitioners need to decide which episodes are more important than others for the software being developed and why; for instance, for an e-marketing website the first impression is more important than it is for a work application. This knowledge can then help practitioners suggest more suitable design solutions. Researchers recommend taking the whole spectrum of interaction into account when studying the UX that a piece of software may deliver, in particular when evaluating it [52].

UX research uses the term **context-dependent** (aka. situated) to emphasize that any experience is unique, unrepeatable, and situated in its context [9]. Nevertheless, experiences can be categorized because their essence is the same, i.e. they connect to essential human needs or be-goals [8]. UX is also **worthwhile** (aka. positive) [8] which emphasizes that UX practices should focus on value and creating desirable experiences and not merely preventing negative ones [8]. On the contrary, usability practices have traditionally focused on removing problems, frustration, and stress (i.e., negative) [8, 9]. However, as Hassenzahl emphasizes, removal of dissatisfaction does not necessarily lead to a good UX.

2.5 From User-Centered Design (UCD) to Experience-Centered Design

As we described before, practitioners are recommended to take into account and apply certain principles and practices (*UX principles and practices*) in order to increase the likelihood of delivering a good UX through the developed software [9, 53]. Practices can be seen as 'what' activities practitioners shall perform in their daily work while principles are the 'why' (the motivation) behind these practices. *Tools and methods* specify in more details 'how' these practices shall be performed to satisfy the principles [13].

Methods impose structure on practices with the goal of making them systematic and ultimately more likely to be successful [13]. Examples are survey, questionnaire, mind mapping, field study, cognitive mapping, design studio [54]. *Tools* are computer-based programs or analog means that assist practitioners in performing various practices and are often designed to support particular methods [13]. Like methods, tools are intended to make the work of practitioners more systematic [13]. Examples are persona, eye-tracking programs, visual design and prototyping tools, Attrakdiff (a specific type of satisfaction questionnaire), and mind-mapping programs.

To the best of our knowledge, the current UX literature does not offer a well-compiled list of UX principles, practices, tools, and methods; still, scattered examples of them, can be found in the literature. In summarizing the studies below, we have used our own understanding and judgment to assign suitable terminologies (principle, practice, tool, and method) even when the terms were missing or different terms were used in the studies. These studies often do not define such terms, or even refer to 'what' practitioners should perform, 'why' and 'how' without labeling them with terms such as the ones defined above. Hence, it is not possible to compare and contrast their views to our understanding and definition of the above terms. Still, where applicable, we have commented on the usage of the terms in various studies.

Hassenzahl [8], for instance, suggests that practitioners should *start the design of products by identifying the particular experiences they aim to deliver through the product.* Hassenzahl, however, does not use any label to refer to his suggestion which in our view is a practice. To clarify this practice, he exemplifies a project in which to satisfy the business need of " bringing knowledge to the society", the designer has focused on 'curiosity' and 'surprise' and used a supermarket as the unexpected context in which shoppers would occasionally hear different pieces of fun facts previously recorded on the devices. Hassenzahl uses his model to clarify the connection of different types of goals and suggests practitioners should then *refine the identified abstract be-goals (in the example for instance: make the user curious and surprised) to more concrete do-goals (hide the product in unexpected locations) and product quality characteristics (the played audio volume shall be low so it resemble a whisper to increase the curiosity of the users)* [53]. He proposes that as tools to support the above practice (identify the be-goals or particular experiences), practitioners can apply existing categories of human psychological needs such as the top-ten needs categorized by Sheldon et al. [55].

As another example, Wright and McCarthy [9], similarly to Hassenzahl, suggest practitioners to first *focus on the particular experiences (identify the be-goals)*, and then *create design solutions to reflect these be-goals*. They refer to this process as *experience-centered design* which, as they highlight, has its roots in User-Centered Design (UCD) (elaborated below). Wright and McCarthy seem to have used the terms principle and practice with a similar meaning as ours. They highlight three main principles for the experience-centered design process: *understanding the end users and the use context*, *meaning-making instead of goals, tasks, and human cognition*, and *enhancing users' experience and improving their lives.*

In order to satisfy the above principles, Wright and McCarthy [9] propose applying *story* and *dialog* in the process. Persona, scenarios, and pastiche are the recommended tools that all use a textual narrative to document and communicate the UX (in form of stories). Wright and McCarthy also suggest using methods such as experience prototyping [56] and drama and role play [57] to engage the practitioners more directly with the needs and desires of the users (through applying both stories and dialog). Other recommended methods to support experience-centered design process include generic ones such as interview, observation, focus group, and diary studies [9].

In their study on UX evaluation, Law et al. [31] highlight that practitioners should *identify relevant UX measures that reflect users' perception about both hedonic and pragmatic aspects of UX*. Law et al., however, do not use any label to refer to their suggestion which in our view is a practice. Hartson and Pyla [42] suggest applying *UX target tables* (including UX measures and metrics) as a tool to support the above practice. Currently, the most efficient and feasible approach to measure perceptions and emotions of users is to directly gather their opinions and let them express themselves [31]. This is often performed through methods such as questionnaires or scales and tools to support them: e.g., AttrakDiff, Self-Assessment Manikin and Affect Gird [50]. However, to gain reliable results, practitioners need to gather responses from a statistically significant number of heterogeneous users [31].

As we saw, often these UX principles and practices have their roots in the UCD process [58]. UCD (also known as Human-Centered Design) is a design approach that emphasizes involving the end users in the design process in order to develop software systems that are more usable and in general provide better experiences to the end users [58]. UCD is not a substitute for software development processes, rather complements them with the following principles (the term that seems to have been used with a similar meaning as ours) [38,59]: *multidisciplinary design team*, *understanding users and context*, *active user participation in the design process*, *early prototyping*, *continuous evaluation*, and *iterative and holistic design*. As depicted in Figure 1, the UCD process is an iterative design process consisting of five steps [58]: *plan the human-centered process*, *understand and specify the context of use*, *specify the users and organizational requirements*, *produce design solutions*, and *evaluate design against requirements*. In our terminology, these steps are in fact practices (i.e. 'what') that practitioners need to perform to satisfy the principles of UCD. Similarly, Gulliksen et al. [59] propose a collection of principles for the UCD process and to introduce these principles, they list various practices related to each principle as follows. Gulliksen et al. seem to have used the term principle with a similar meaning as ours while they have used the term activity to refer to what we call practice.

- *User focus*: create and maintain a focus on the users' needs instead of a technical focus. Methods such as contextual inquiries and task analysis support this principle [42].
- *Active user involvement*: involve representative users early and continuously throughout the life-cycle of the software.
- *Evolutionary systems development*: perform an iterative and incremental development and ensure that the solutions are continuously evaluated together with the user representatives.

- *Simple design representations*: ensure that the proposed design in each iteration is understandable by users and all other stakeholders. Prototypes and sketches, for instance, may be used for such a purpose.
- *Prototyping*: prototype the design ideas (early and continuously).
- *Evaluate use in context*: steer the development based on usability and design goals and in collaboration with users, test the design solution against the usability goals.
- *Explicit and conscious design activities*: prioritize design practices to ensure that the GUI does not just 'happen' due to coding or modeling.
- *A professional attitude*: establish a multidisciplinary team who takes care of analysis, design, and development in a collaborative and professional manner. Ensure that usability practitioners are part of such a team.
- *Usability champion*: involve usability practitioners, with enough authority to secure UCD, early and continuously in the life-cycle of the software.
- *Holistic design*: take the whole use situation into account when designing software (e.g. work/task practices and work/task organization, user interface and interaction, online help, manuals, user training, health and safety aspects, etc.).
- *Processes customization*: customize the UCD process based on the context and needs of each organization as there is no 'one-size-fits-all' process.
- *A user-centered attitude*: ensure that all of the stakeholders (both the team and the customers) are aware of and committed to the importance of usability and user involvement.

The HCI community has developed and proposed various tools and methods to support different steps in the UCD cycle although with a primary focus on usability [38, 54, 59]. Maguire [38], for instance, presents a variety of well-known methods to support the UCD process. Examples are: diary keeping, task or function mapping, stakeholder analysis, brainstorming, and participatory evaluation.

UCD practices and methods are often associated mainly with usability, nevertheless, they can also be used (and are recommended to be used) having the whole UX in focus [60]. Yet, empirical data shows that although an ideal UCD process should focus on the overall UX, this aspect of UCD is often ignored in practice [61]. Furthermore, whilst research on UX emphasizes the hedonic aspect of software use, practitioners who apply UCD still mainly focus on functional and usability issues (i.e. merely the pragmatic aspect of UX) [7]. Hence, simply applying generic UCD practices without careful attention to and being committed to addressing differences between UX and usability does not necessarily result in delivering a good UX through the software. For instance, methods such as focus groups or user interviews can be applied in the context of UX. However, for these methods to be effective in the case of UX, when applying these methods practitioners should take the main UX characteristics into account, as we later elaborate in our findings.

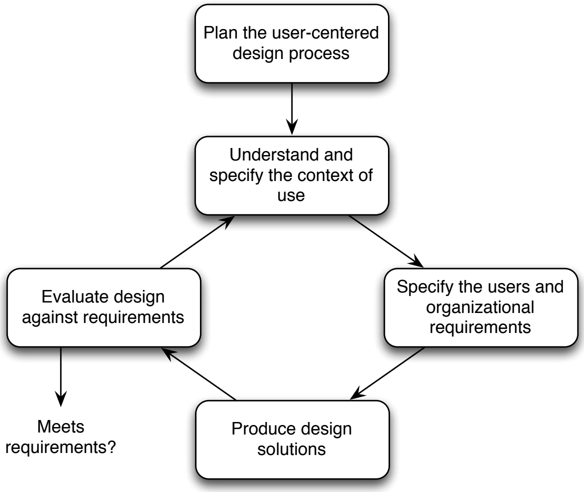
Plan the user-centered design process
Understand and specify the context of use
Specify the users and organizational requirements
Produce design solutions
Evaluate design against requirements
Meets requirements?

Figure 1: User-Centered Design (UCD) process

3 Related Work

Empirical studies show that software organizations often deal with various challenges and success factors in their work with quality characteristics, in general, and UX and usability, in particular.

For instance, Berntsson Svensson et al. [20] found that if practitioners lack knowledge and understanding about software quality characteristics, they tend to undervalue and ignore these characteristics during development. Berntsson Svensson et al. also found that practitioners are more likely to dismiss those characteristics that are considered less important. According to their findings, usability is one such characteristic that is perceived to be less important, hence dismissed in projects. Similarly, Karlsson et al. [62] show that quality issues are often perceived to be less important than functional issues and that practitioners find it difficult to deal with dependencies between quality requirements or between quality and functional requirements.

Research shows that, in general, practitioners find it more difficult to perform requirements and testing activities for quality characteristics than for functionality [20, 21, 36]. For instance, they find it more difficult to document quality requirements in a measurable way or handle their dependencies to functional requirements [19, 20, 62, 63]. Borg et al. [63] report that lack of competencies to document and or to test quality characteristics often leads to ignoring them in projects.

Similar problems have also been discussed in usability literature, for instance, Bak et al. [64] report that developers' minds are set more on the programming aspect of the product than its usability and often misunderstand the term "usability evaluation" and relate it to functionality. Gulliksen et al. [65] report that limited awareness in different levels of organizations can lead to down-prioritizing usability. They, therefore, suggest increasing knowledge and awareness about usability among different stakeholders and in various levels of organizations, in particular among management, in order to improve usability integration in organizations. In addition, practitioners find it challenging to document measurable usability requirements [65,66] although failing to include usability in requirements documents may result in ignoring these requirements later in testing [28].

Limited access to competencies and unclear responsibilities are also reported as challenges to usability integration. For instance, Rosenbaum et al. [26] report that lack of usability professionals is one of the main obstacles organizations face concerning usability. Boivie et al. [67] show that even in cases when organizations have access to right expertise, usability professionals are not sure about their responsibilities and are uncertain as to how they shall contribute to the projects. Chamberlain et al. [68] report that power struggles arise in organizations as designers within a project defend their discipline in response to the decisions made by developers and vice versa.

Despite the differences between UX and usability, only a limited number of empirical studies have so far analyzed the implications of these differences on the day-to-day work of practitioners. One example is the study by Vermeeren et al. [30] that compares the challenges in evaluating usability and UX. They argue that some of UX evaluation methods need to be further improved and developed for better use in the industry. According to Vermeeren et al., there is still a lack of suitable methods for evaluating UX in earlier development phases or in the period before actual use (i.e. anticipated use). They also highlight that current methods are not often practical because they need special expertise, are time-consuming, or their data analysis is hard. Similarly, Isomursu et al. [17] discuss that practitioners face difficulties when performing UX practices (compared to usability practices) because they do not have access to tools and methods to objectively measure UX.

Through an interview study, Law et al. [31] explored the basic question of whether UX is measurable. They report that their interviewees expressed skepticism and ambivalence towards specific UX measures even if attitudes were more positive overall. They note that practitioners show opposing views on whether UX can or should be divided into composing elements, or whether it needs to be considered or measured as a whole. The participants in the Law et al.'s study emphasized they need to use a variety of media (e.g., video, TV, social media) to develop the required prototypes for measuring UX and that they often even need more than one such prototype to gather enough input for design. Their practitioners also argued that UX measures are essentially prone to fading and fabrication or that there is a lack of means to measure the exact emotion of users at each moment [31]. Law et. al., therefore, divide challenges concerning the interplay between UX evaluation and software development into the following categories:

- *Theoretical challenges* which include: (i) it is difficult (even if possible) to measure UX in a holistic way and breaking it down into components seems not to be an ideal solution either, (ii) memorized experiences are prone to fading and fabrication which influences the response of the UX evaluation participants, (iii) UX measures are highly sensitive to timing and nature of tasks which makes it difficult to design UX evaluations.

- *Methodological challenges* which include: (i) UX and non-UX practitioners have different preferences for qualitative and quantitative data, (ii) UX evaluation is resource demanding and requires a large number of heterogeneous users to be involved in the evaluations, (iii) for eliciting authentic user experiences, there is a need for sophisticated prototypes.

- *Practical challenges* which include: (i) there is a lack of knowledge in exploiting feedback on UX for future system development in particular from qualitative data, (ii) there is a lack of standard UX metrics which makes redesign decisions prone to personal biases, (iii) it is difficult to package UX measures for decision makers and speak their language.

The work of Law et al. [31] is duplicated in the context of the Latin American software development industry by Gerea et al. [32] that conclude practical aspects such as cost and time play an important role in whether or not practitioners measure UX. Other challenges reported by Gerea et al. include limited access to the end users and lack of knowledge and experience in UX measurement similar to what Vermeeren et al. [30] found.

Lallemand et al. [69] investigated practitioners' understanding about the concept of UX, including their opinion on a number of UX definitions. This was achieved through an international survey study with 758 participants including practitioners and researchers with both UX and non-UX-related backgrounds. They provide an overview of how practitioners currently perceive the concept of UX and its importance in software development. They also highlight the challenge of *the low impact of UX research on industry* and stress that various research findings about UX (e.g. its relation to usability) are not still well received in the industry. They, therefore, emphasize to better integrate UX theories in the industry and educate students on UX research.

Alves et al. [34] investigated how UX evaluation is performed in the industry (i.e., by whom, in what phases of software development, and using what tools and methods). According to their data, in around 50% of the cases, UX evaluation is performed without involving the end users, and often evaluators 'assume' what the perception of users will be. Alves et al. [34] also report that sometimes evaluations are performed by software developers that do not necessarily have the required competencies. Also, often tools and methods are selected based on cost rather than suitability for the project. Alves et al. used a list that includes mainly usability-specific evaluation tools and methods and lacks UX-specific ones (e.g., Attrakdiff questionnaire [70][3]). This can introduce a risk to their data because practitioners might have preferred applying a generic method such as a questionnaire for evaluating UX, without necessarily acknowledging that such generic methods may purely produce usability-related data if not used with specific attention to UX characteristics, and in particular, the non-task-related aspect of use [71].

Studies also show that the perception of UX is generally different in academic and industrial contexts. Väänänen et al. [7] report that practitioners still focus more on functionality and usability than UX. Similarly, in Kuusinen et al's study [35] in a large software organization, ease of use and efficiency were the most often reported sources of good UX. Hence, these studies conclude that while academia concentrates more on the hedonic aspect, the industry focuses more on the pragmatic aspect.

[3] see `http://www.allaboutux.org` for a list of UX-specific tools and methods

As we saw, those studies that explicitly take differences between UX and usability into account, mainly focus on how the concept of UX is perceived in the industry or on evaluation activities and the role of UX measures in challenges practitioners face. They, therefore, do not sufficiently discuss other topics related to UX integration (e.g. communication and collaboration between UX and non-UX practitioners). In addition, these studies do not often analyze UX integration challenges and success factors in relation to those previously reported for usability or other software quality characteristics. They also often present a snapshot of the integration in organizations and do not analyze how organizations have moved from only developing user interfaces to also paying attention to usability and more recently UX. Such analyses can provide a more comprehensive understanding of UX integration, its related challenges and success factors and can also help to learn from integration efforts in other contexts.

4 Problem Statement and Main Objectives

Our main research aim was to facilitate and improve the current state of UX integration in the software industry. This aim was defined in form of the following research problem: *How can software companies integrate UX principles and practices into their development processes and organizations?* The research problem was then broken down to a number of research questions as described below.

Addressing the above research problem first requires improving the software community's understanding of current state of UX integration in software companies and also gaining a better understanding of challenges and success factors in UX integration in the industrial context. We, therefore, created the following research question: *What challenges do practitioners face in integrating UX practices into software development processes and organizations?*[4]

Considering that UX can be seen as a type of software quality characteristic that shares some traits with usability, we found it important to understand how UX integration challenges and success factors relate to the ones previously reported for other quality characteristics, in particular, usability through defining the question: *How do UX integration challenges relate to challenges in handling software quality characteristics, in particular, usability?*[5] We believed the knowledge that could be gained by addressing the above question could help the community to learn from, apply, or customize and extend the existing principles, practices, challenges, and success factors also in the context of UX integration. These two research questions were addressed in our first study.

The findings of the first study drew our attention to the fact that the software community has a general lack of understanding regarding the relation

[4] Slight language improvement is applied here by using 'in' instead of 'of' which was used in the original research question in Paper I.

[5] The term 'integration' is added to this research question here and was not part of the question in Paper I. After publishing our first study we acknowledged that the second phrase better communicates the meaning we had in mind. Slight language improvement is also applied here by using 'in' instead of 'of' which was used in the original research question in Paper I.

of UX to other software quality characteristics. Hence, we found it important to also compare and contrast UX as an emerging software quality characteristic to other more traditional quality characteristics but also software functionality. We, therefore, defined another question to explore this matter: *What is the relation between UX and other software quality characteristics, in particular, usability?*

Furthermore, the findings of the first study showed that UX integration challenges could be of both technical and organizational nature. We, therefore, decided to investigate these categories of challenges more in depth through a case study. The knowledge that we had gained through literature studies and our observation of the changes in the organizations we collaborated with motivated us, in this next study, to not merely focus on challenges and success factors but also the events inside and outside the organization that impacted the integration. In our view, understanding these events could provide a more comprehensive and realistic image of the UX integration in its realistic context: ever-changing organizations. The case study, therefore, was designed to address these research questions: *How does UX integration unfold over time within the context of an organization? And, what are the main intertwining events that impact UX integration as it unfolds?* and *How do organizational and technical issues enable or prohibit integrating UX principles and practices into software development processes and organizations? And, how does the nature of UX impact these issues?*.

In our case study, we also explored the research question of *What can UX practitioners and researchers learn from software process improvement body of knowledge?* after we realized that our findings underline the similarities between UX integration and Software Process Improvement (SPI). This notion was further investigated in our last study where we took a more theoretical approach to SPI nature of UX integration.

As Table 1 summarizes, we have addressed the above research questions in the six papers included in this thesis. Based on these research questions, the work of this thesis can be positioned in the intersection of the two fields of Human-Computer Interaction (HCI) and Software Engineering (SE), and more specifically, the intersection of the two areas of UX and Quality requirements in these two fields (see Figure 2).

5 Research Design

We started this thesis work with an 'explorative' purpose [72]: learning more about the current state of UX integration in the software industry, in order to gain new insights and generate ideas and hypotheses for later stages of our research which we achieved through interviewing practitioners in different companies. At this stage, we did not restrict ourselves to focus on any specific areas of challenges rather aimed to gather as much variety of these challenges as possible. After generating a wide range of challenges, we then planned to investigate them in more depth to also 'describe' and 'explain' how and why they appear in organizations and how they impact UX integration. In addition to challenges, we also aimed to gather the factors that could enable UX integration in software companies. The descriptive and exploratory purpose [72]

	Papers					
Research Questions	**I**	**II**	**III**	**IV**	**V**	**VI**
RQ1: What challenges do practitioners face in integrating UX practices into software development processes and organizations?	✓					
RQ2: How do UX integration challenges relate to challenges in handling software quality characteristics, in particular, usability?	✓					
RQ3: What is the relation between UX and other software quality characteristics, in particular, usability?		✓				
RQ4: How does UX integration unfold over time within the context of an organization? And, what are the main intertwining events that impact UX integration as it unfolds?			✓			
RQ5: How do organizational and technical issues enable or prohibit integrating UX principles and practices into software development processes and organizations? And, how does the nature of UX impact these issues?				✓		
RQ6: What can UX practitioners and researchers learn from software process improvement body of knowledge?					✓	✓

Table 1: The research questions addressed in each paper

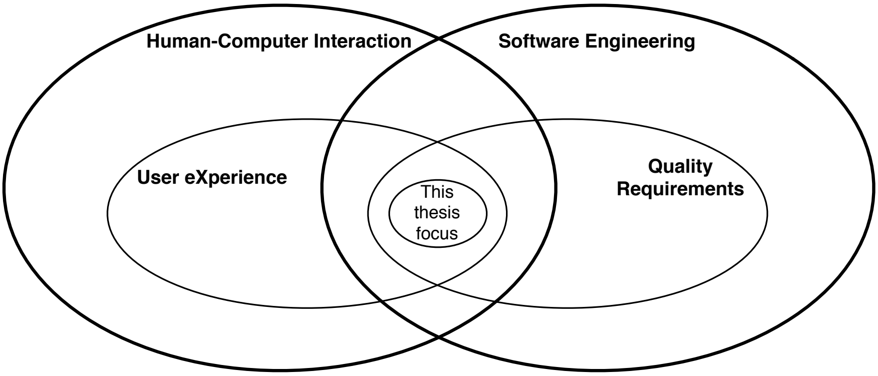

Figure 2: The focus of this thesis (main research aim and contributions) in relation to the two fields of SE and HCI

of our research at this stage was achieved through a case study.

Some parts of our research had an ‘improving’ purpose [72] where we aimed to identify and propose solutions at least to a number of challenges we had identified. Part of the challenges, identified in our first interview study, related to inadequate UX knowledge and awareness among practitioners and that current UX models are commonly not practical nor well integrated into existing Software Engineering (SE) models and concepts. Therefore, in **Paper II** we presented a conceptual UX-aware model of requirements for software development practitioners. Through **Paper I** and **Paper III** and **Paper IV** we also identified that although UX integration is a type of organizational change, more specifically, a type of Software Process Improvement (SPI), practitioners often fail to benefit from the existing body of knowledge in these two fields. Therefore, in **Paper V** and **Paper VI**, we approached UX integration from an SPI perspective and showed how practitioners can benefit from existing SPI principles and practices to enhance UX integration in their organizations.

All of the papers included in this thesis rely on empirical data: data gathered directly from industry. However, Paper I, Paper III, Paper IV, Paper V have a stronger empirical focus. Paper II and Paper VI, on the other hand, although being theoretical to some extent, also emerged from the result of our empirical studies, therefore, in our view, can be also considered empirical. In addition, all of these studies are qualitative which means they apply data gathering methods that result in qualitative data (text, descriptions, pictures, diagrams as opposed to numbers) [72]. These studies also apply data analysis methods that through sorting and creating themes and categories generate knowledge and insight [73]. Our choice of empirical research is motivated by the nature of our research problem. In order to be able to help practitioners enhance the state of UX integration in their companies, a natural step is to first understand how these practitioners perceive the current state of practice (including both challenges and success factors).

One could argue that quantitative studies could also generate similar insights. Admittedly, such studies are often stronger in identifying causal relationships and prioritizing the findings (e.g. identifying the most common challenges or most influential success factors). Their shortcoming, however, is that they do not necessarily provide enough explanation and description of the gathered data (e.g. why a certain challenge is more common). In addition, quantitative studies require a statistically significant amount of data to be able to draw such causal or prioritization conclusion while at early stages of our research (2012) we considered it to be a likely threat to our research. We found it likely not to be able to gather enough amount of data mainly based on our experience and knowledge of the Swedish software industry that, in our view, was still young in integrating UX principles and practices.

While our first two papers report on the results of our interview studies, the rest of the papers are based on a case study performed in a Swedish software development company. Case study is an empirical research methodology that is used in order to study a phenomenon within its real-life context [73]. This case study provided rich empirical data on UX integration and its challenges and success factor.

We used various data gathering methods in our research work, including interview, observation, workshop, questionnaire, and document analysis. Based

Paper	Type of Research	Applied Methods
I	Empirical research	Thematic analysis, Interview
II	Empirical research	Literature analysis, Interview
III	Empirical research	Case study, Grounded theory, Interview, Observation, Document analysis
IV	Empirical research	Case study, Grounded theory, Interview, Observation, Document analysis
V	Empirical research	Case study, Workshop
VI	Empirical research	Literature analysis, findings from paper III and IV

Table 2: Type of research and methods applied in the studies included in this thesis.

on the goals and scope of our studies, we applied these methods to different extents; however, interviews, observations and document analysis were most used and are, therefore, described below in more details. A mapping between different studies, their type of research, and used methods is summarized in Table 2.

In our work, we focused on identifying and understanding UX integration challenges and success factors from practitioners' perspective, hence, found interviews the most suitable method to apply. In our interviews, we engaged in a dialog with the interviewees about UX and UX integration in their organizations among other things. Surveys [72] could also be used for this purpose, however, we not only were interested to identify these challenges and success factors but also to understand why they appear in different organizational contexts and how they may facilitate or prohibit integration. Such information could best be gathered in a dialog with practitioners, where researchers could ask follow up questions to better understand the mentioned challenges and success factors.

In all of our studies, we conducted semi-structured interviews [72] to collect more of the interviewees' viewpoints. We prepared a set of initial questions on the topics we needed to ensure are covered in the interviews. We also asked follow up or improvised questions during the interviews and based on the interviewees' responses.

In the case study, we also used observations [74] to gather information about UX integration in the case company and integration efforts being performed. Observations may provide a deep understanding of the phenomenon under study, a disadvantage, however, is that they provide a substantial amount of data that makes the analysis time consuming [74]. We observed various meetings in order to gather information about the state of UX integration in the company and the beliefs and attitudes of various groups of practitioners towards UX integration. These observations helped us gather rich data on the activities and also the dynamic between UX and non-UX practitioners.

To get a better understanding of UX practices and their impact on projects in the case company, we also analyzed organizational charts, requirements documents, UX guidelines, and GUI mockups. This type of data gathering is called *archival data analysis* and is an indirect source of information for researchers [74]. In the case company, archival data was used to triangulate the data gathered through interviews and observations.

In our interview studies, we applied *thematic analysis* to generate patterns from data as described in Braun and Clarke [75]. To encode and analyze the interview data, we *segmented* the interview transcriptions into meaningful paragraphs or sentences in a way that each of these segments presented one concept. We then *coded* these segments [75]. We used Microsoft Excel to document the coded data where each interview transcript was recorded on a separate sheet. Segments of this transcript were recorded in separate rows and different codes were assigned to each segment in separated columns, following that segment. After coding, categories of challenges, i.e. themes, emerged as we put together similar concepts presented in the coded segments. A mind-map of challenges and categories was created to better identify and visualize the relations.

In our case study, we used Grounded Theory (GT) [76] as the overarching methodology to study the data and to drive the data gathering and analysis activities. GT is an established and credible methodology in socio-logical disciplines (e.g. psychology). We simultaneously performed data gathering and analysis until we achieved theoretical saturation [76]. We also performed constant *memoing* while coding the data. GT has two main different variations, Glaserian (aka. classic) [77] and Straussian [76] that differ in their steps and principles (for a comparison please see [78]). We picked Straussian version and followed Strauss and Corbin's recommendations for performing GT [76] including the following steps:

- *open coding* is the step in which the concepts are identified in the data. For this purpose, the raw data (e.g., interview transcripts or observation notes) was broken down into manageable analytical pieces. These pieces then were openly tagged with codes (i.e., concepts and categories) by identifying key points represented in the segment.

- *axial coding* is the process of relating categories to their subcategories. In this step, the generated codes were related to each other via a combination of inductive and deductive thinking to form the main categories (or themes). The main categories however merely describe the data and need to be further developed into a theory.

- *selective coding* is when the core category is selected from the main categories identified in axial coding. The theory is refined and developed through linking the identified categories around the core category.

In this study, we performed several rounds of coding and presented the result of our grounded theory in form of a mind-map of the identified themes of challenges and success factors, and a collection of UX integration principles and practices. Considering the current lack of theories on UX integration, using grounded theory approach, therefore, encouraged us to engage with theories outside UX integration literature. As a further way to evaluate our findings, we put them not only in relation to previous empirical findings on UX and usability integration but also literature on software process improvement [79] and organizational change [80].

6 Overview of The Appended Papers

This section presents a brief summary and the main results from each of the included papers in form of answers to the research questions.

6.1 Paper I: Integrating User eXperience Practices into Software Development Processes: Implications of UX Characteristics

To address RQ1 and RQ2, in our first study, we explored current challenges practitioners face in their UX integration efforts. We designed an interview study and in semi-structured interviews gathered practitioners' viewpoints regarding the current state of UX integration in their organizations. We interviewed 17 practitioners with different backgrounds and occupations from eight software development companies. Their responses were coded and analyzed with thematic analysis.

In answer to RQ1, *What challenges do practitioners face in integrating UX practices into software development processes and organizations?*, we found 8 themes of challenges:

- Theme 1. *Lack of consensus on the definition and construct of UX*: we found that, for instance, practitioners have different understandings or even contradicting views on the concept of UX or that they sometimes consider UX as only equal to usability or interaction design.

- Theme 2. *Lack of consensus on the value of UX*: our findings show that, for instance, in the development processes, UX is often down-prioritized to functions as the cost associated to it is not justified and its value for the business success of the software is unclear to stakeholders.

- Theme 3. *Low industrial impact of UX models, tools and methods*: we found that, as an example, practitioners often gain their UX knowledge ad-hoc and not necessarily through existing UX research and theories. They often also use traditional tools and methods such as interviews and observations in analysis or evaluation of their products without taking the whole concept of UX into account.

- Theme 4. *Focus on the objectively measurable aspects of software*: our findings, for instance, show that due to difficulties in measuring or agreeing upon subjective matters (i.e. the hedonic aspect of UX), they are most often overlooked or down-prioritized in projects compared to functions or objectively measurable quality characteristics.

- Theme 5. *Difficulties in engineering UX-related requirements*: we showed that, for instance, practitioners find it difficult to reach a balance between business, technical and UX-related requirements, or that they often find it hard to elicit, refine, or communicate UX-related requirements within projects.

- Theme 6. *Focus on evaluating functionality and usability, not UX*: our findings show that, as an example, often UX evaluation is performed only informally in projects, or that practitioners find it difficult to evaluate the hedonic aspect of UX, in particular, in comparison to testing usability and functions.

- Theme 7. *Lack of consensus on UX-related competencies and responsibilities*: we found that, for instance, organizations suffer from limited access to UX-related competencies and low management support to gain such competencies.

- Theme 8. *Communication and collaboration gap between UX and non-UX practitioners*: our findings show that, for instance, often in organizations there is a lack of trust and a power-struggle between UX and non-UX practitioners.

Some of these themes are more fundamental and concern the views, attitudes, and knowledge of stakeholders while some others are more tactical (see Figure 2 in Paper I). These themes clearly have a multifaceted and complex set of relations. At a high level, the more fundamental themes can explain the tactical ones. One example is the connection between limited knowledge of UX (Theme 1) with the low impact of UX models in the industry (Theme 3) which in turn can lead to a language gap and communication problems (Theme 8) within software development organizations.

As we mentioned, the current literature on UX integration challenges often discusses them in isolation and does not compare and contrast them to the findings of previous research on usability or other software quality characteristics. Such an analysis of the challenges can help practitioners better understand them, put them into perspective and even, when possible, apply the existing approaches to address previously known challenges to tackle UX integration challenges as well. Hence, in this paper, we also answer RQ2: *How do UX integration challenges relate to challenges in handling software quality characteristics, in particular, usability?*

We found that there are in fact overlaps between these two groups of challenges. For instance, the challenges in requirements and testing are two categories that are common between UX, usability [26, 81], and quality characteristics [82, 83]. More importantly, we found that although at a superficial level challenges in integrating UX and other quality characteristics overlap, they could be differentiated at a deeper level through the main characteristics of UX: *subjective*, *holistic*, *dynamic*, *context-dependent* and *worthwhile*.

We identified that these characteristics have at least 20 implications (i.e. additional difficulties) for the day-to-day work of practitioners, nine of which are unique to our study. These implications impact various aspects of software development and concern not only specific phases such as requirements, design, and evaluation but also the communication and collaboration between different groups of practitioners. Examples of these implications are: *UX practices are perceived to be more visible in the development process than usability practices*, *compared to the case of usability, more power struggles, and disagreements are perceived to rise between UX and non-UX practitioners*, *abstract UX-related needs (e.g., be-goals or emotional needs) are difficult to refine and translate into more concrete ones*, *it is hard to translate abstract UX-related needs to concrete design solutions*, and *it is hard to frame UX in evaluations because of* *all the complex underlying dependencies between its elements* (see Table 4 in Paper I for a complete list of these implications).

The findings of this study, and in particular the presented implications of the UX characteristics, help us explain why practitioners often perceive UX integration challenges to be more severe than for other quality characteristics. These findings can also be one explanation as to why the software industry has a lopsided focus of on usability, in particular, and the pragmatic aspect of UX, in general. Hence, we emphasize that to achieve a successful UX integration, practitioners need to consider the inherent differences between UX and other quality characteristics and explicitly take the five UX characteristics and their implications into account in the integration efforts.

6.2 Paper II: A Conceptual UX-aware Model of Requirements

Part of the challenges we identified in answer to RQ1 and RQ2 relate to inadequate knowledge and awareness of UX and that current UX models are commonly not practical nor well integrated into existing Software Engineering (SE) models and concepts (Theme 3. Low industrial impact of UX models, tools and methods). This motivated us to propose a conceptual UX-aware model of requirements for software development practitioners to address such challenges at least to some extent.

Paper II reports on the study we performed to address RQ3: *What is the relation between UX and other software quality characteristics, in particular, usability?* We followed the technology-transfer-model [84], developed a UX-aware conceptual model for requirements in close collaboration with practitioners, and validated it through interviews with 12 practitioners and researchers. The results of the interviews showed that the model can raise practitioners' knowledge and awareness of UX in particular in relation to requirement and testing activities and can also facilitate UX-related communication among stakeholders with different backgrounds.

The model (Figure 1 in Paper II) aims to enhance UX knowledge and awareness of both UX and non-UX practitioners from a requirements perspective. It introduces the concept of *UX requirements* and puts it in relation to two other requirement types: *objective Quality Requirements* (objective QRs) and Functional Requirements (FRs). We aimed to clearly communicate a number of differences between these three categories of requirements to practitioners. For instance, the model highlights that the role of *human perception* (therefore, subjectivity) and the level of *abstraction* increases as we move upwards in the model from functional to objective quality requirements and then UX requirements.

As this study highlights, it is of critical importance for the software development community to understand and acknowledge the differences between UX and other quality characteristics (as we discussed before). More specifically, it is important that practitioners acknowledge the implications of these differences for requirements and testing practices as the model communicates. Otherwise, it is likely that UX as a quality characteristic is not sufficiently handled in these two important phases of software development which creates a risk of under-prioritizing or even ignoring UX in development processes as it has been the case for other quality characteristics [83].

6.3 Paper III: Integrating UX Principles and Practices into Software Development Organizations: A Case Study of Influencing Events

In order to further analyze the UX integration challenges identified in Paper I, we designed a case study in a Swedish software development company to investigate more than two decades of their integration efforts through interviews, observations, document analysis, and workshops. Another main motivation behind this case study was that still little is known about the transition that companies go through from only designing Graphical User Interfaces (GUI) to also paying attention to usability and more recently UX. Through this study we aimed to address RQ4: *How does UX integration unfold over time within the context of an organization? And, what are the main intertwining events that impact UX integration as it unfolds?*

UX integration, like other organizational changes, can include a mixture of planned and emergent initiatives, and is influenced by various intertwining events; not only those that reside inside the organization but also those external to it. Here, by event, we mean any decision, activity, action, or circumstance that contributes to changes in the organization. We identified at least three different categories of these events that may influence UX integration:

- external and internal: events that reside outside or within the borders of the organization
- direct and indirect: events that explicitly and directly change UX integration in the organization or those that influence UX integration although this is not their main purpose
- planned and emergent: planned (aka. top-down) change initiatives designed to influence UX integration or those grass-root (aka. bottom-up) change initiatives performed to influence UX integration

Examples of these events are: *changing the development process from waterfall to agile* (internal, indirect), *technological advances in the field of software development* (external, indirect), *assigning a new head of R&D to improve UX integration in the company* (internal, direct, planned), *establishing UX meetings by a number of UX advocates in the company* (internal, direct, emergent).

Despite our findings on the role of external, emergent and indirect events on UX integration, the current literature mainly focuses on internal, planned, and direct events for improving UX integration in organizations (e.g. [65, 85, 86]). Our study shows that achieving a sustainable UX integration (an integration not only successful short-term but also maintained over time), requires investigating and adjusting to the impact of indirect events on integration as well as directly applying and integrating UX principles and practices. In addition, a sustainable UX integration requires investigating and reflecting on both emergent and planned changes in the organization. Practitioners also need to take into account the role of external as well as internal events in enabling or prohibiting UX integration. We also show that different decisions that are made outside the authority of UX practitioners have an inevitable impact on enabling or prohibiting UX integration. In addition, we found that for a successful integration, practitioners need to explicitly consider and address the characteristics of UX, otherwise, the UX integration efforts may have a lopsided focus on the pragmatic aspect of UX, and consequently, leave the hedonic aspect unaddressed.

Based on our findings, we identified 4 lessons learned and 5 pitfalls companies should consider to go beyond GUI design and usability to also address UX. These lessons learned and pitfalls mainly concern the organizational aspects of UX integration. Here, by organizational aspects we mean those aspects that concern practitioners and their context of work, the organizations. Organizational aspects concern how and in what structure and setting various practitioners work together in an organization in contrast to what specific practices they perform, or what tools, methods, and processes they use in their work, what we call the technical aspects.

The identified pitfalls cover topics such as knowledge and awareness, attitudes, organizational culture, organizational learning, and communication and collaboration between UX and non-UX practitioners (for a list of these pitfalls please see Figure 3 in Paper III). To avoid these pitfalls, we proposed 12 practices and one principle that practitioners need to apply in their integration efforts. Examples of such practices are *disseminate the differences and similarities between UX, usability, and GUI design in the organization*, *give mandate to UX advocates to extend their activities to address the hedonic aspect of UX as well*, and *raise awareness in the organization about the relation of UX to service design, business strategy, and innovation*.

Similarly, the identified lessons learned underline the importance of the organizational aspects of integration and concern four topics: influencing events, success factors and challenges, resistance to UX integration, and UX integration as an organizational change. To further support practitioners, we refined these lessons learned into four UX integration principles and five UX integration practices (summarized in Figure 2 in Paper III). Examples of these principles are: *UX integration is a type of organizational change* and *resistance to UX integration may be shown by both UX and non-UX practitioners*. The practices, for instance, include *investigate and reflect on events outside the organization that may indirectly influence UX integration*, and *apply existing guidelines on organizational change to plan, execute, and support UX integration in the organization*.

6.4 Paper IV: Integrating UX Principles and Practices into Software Development Organizations: A Case Study of Challenges and Success factors

Paper IV reports on the result of our case study with a focus on challenges and success factors that over time prohibited or enabled UX integration in the company. This paper addresses RQ5 (which admittedly extends RQ1): *How do organizational and technical issues enable or prohibit integrating UX principles and practices into software development processes and organizations? And,* *how does the nature of UX impact these issues?*

We found that UX integration challenges and success factors can concern technical or organizational aspects of integration. The identified challenges and success factors belong to three main themes (detailed in Figure 2 in Paper IV):

- *Beliefs and attitudes concerning software quality and UX:*
  This theme concerns practitioners' understandings, views, and attitudes. Our findings, for instance, show that the way practitioners perceive the value of software quality in general and UX in particular, influences their attitudes towards UX integration. A more specific example of a similar concern is how practitioners perceive the importance of hedonic versus pragmatic aspect of UX.

- *Business alignment of UX integration:*
  This theme concerns value delivery and the importance of the users' needs and personal goals in comparison to the customers' needs and business goals. While the former is the heart of UX practices, the latter has traditionally been the focus of business development practices. To achieve a sustainable UX integration, practitioners need to pay attention to the position of UX in the business strategy and as a competitive edge. They also need to take into account the role of UX integration initiatives and UX practices in minimizing time to market and increasing value for the development organization.

- *UX practices and responsibilities:*
  This theme concerns the often collaborative nature of UX practices and how the decisions and actions of various stakeholders may directly or indirectly influence the delivered UX through the developed software. One of our important findings was that power struggles can happen not only between UX and non-UX practitioners but also among UX practitioners, as bottom-up and top-down UX integration initiatives intersect. To overcome such a challenge, organizations should, for instance, better investigate previous emergent changes concerning UX integration and better align them with planned ones.

This study confirmed the findings of Paper I yet additionally found richer data on the organizational aspects of integration such as the role of culture and business model and strategy on the integration. Our findings also extend previous research [65, 87] that points out the culture and its role in UX integration but does not often provide a detailed view of it. In relation to culture, we identified at least four factors that may impact forming a culture for or against UX integration in organizations: (i) the nature of UX, (ii) the evolution of UX from GUI design, (iii) the business alignment of UX integration, and (iv) previous experiences with UX integration.

In addition, this study showed that the identified challenges and success factors are dynamic and their role in enabling or prohibiting UX integration changes over time. In this study, we refined the identified challenges and success factors into 12 UX integration practices (Figure 3 in Paper IV) that practitioners should perform to increase the likelihood of achieving a sustainable UX integration in their organizations. These practices include a variety of activities that mainly concern the organizational aspects of UX integration and concern culture building, knowledge and awareness, and change management. The practices, for instance, include: *ensure that all internal stakeholders are informed about and consulted when introducing UX- related roles and responsibilities*, *ensure that UX vision is part of business model and strategy*, and *explicitly separate UX practices from 'UX integration' practices, and allocate recourses to them.*

6.5 Paper V: Cross-Section Evidence-based Timelines for Software Process Improvement Retrospectives: A Case Study of User eXperience Integration

In Paper V, we approached UX integration in the case company from an SPI perspective. The aim of this study was to address RQ6: *What can UX practitioners and researchers learn from software process improvement body of knowledge?*

One of the findings of our case study was that practitioners in the case company did not have enough reflection on their past usability and UX integration efforts. Therefore, they had fewer opportunities to learn from their past experiences to increase their chance of achieving a sustainable UX integration. Hence, following the recommendations in SPI literature, we performed a retrospective meeting at the company to reflect on their decade of integration efforts.

We supported the meeting by a pre-generated timeline of the main activities in the organization in a different manner than common project retrospective meetings in SPI where such preparation is not required. This approach is a refinement of a similar approach that is used in Agile projects and is shown to improve the effectiveness of the meeting and decrease memory bias of the participants [88]. We hypothesized that this method can be useful in the context of UX integration which showed to be plausible based on the practitioners' views gathered through questionnaires. Our findings show that this method could be useful for reflecting on, learning from, and coordinating UX integration activities.

6.6 Paper VI: Stakeholder Involvement: A Success Factor for Achieving Better[6] UX Integration

As with Paper V, this paper focuses on RQ6: *What can UX practitioners and researchers learn from software process improvement body of knowledge?*. Through this study, we highlight that the SPI body of knowledge can also inspire UX integration research and practice with regards to stakeholder involvement. Here, by stakeholders we mean UX and non-UX practitioners and management that all play an important role in achieving a sustainable UX integration.

Research on SPI is more mature than UX integration research and practitioners have access to ample SPI guidelines (collection of principles and practices). These guidelines, however, need to be customized to better suit UX integration and, in particular, accommodate the five unique characteristics of

[6] Here, by 'better' we mean that the state of UX integration shall be improved compared to its current state in the software industry

UX. As our previous studies showed, these characteristics have various implications for day-to-day work of practitioners and also integration. In our view, simply applying SPI guidelines does not satisfy the particular needs of UX integration as these guidelines are generic and do not directly address UX or any other specific software quality characteristics for that matter [89]. Hence, this paper aimed to create and propose customization of a number of SPI recommended practices to support UX integration efforts.

As our first step, we investigated how each UX characteristic may impact the daily work of UX and non-UX practitioners (in requirements, design, or evaluation phases) and also on stakeholder involvement which is vital for integration. We used the findings of our previous studies in order to prepare a list of such implications. We then created a list of *UX integration practices* that practitioners need to perform in order to enhance UX integration in their organizations. These practices focus on the organizational aspects of UX integration and, in particular, stakeholder involvement. Examples of these practices are: *promote the benefits of improving the UX delivered through the products and services among the staff*, *ensure that management provides strong leadership and support for UX integration and UX practices*, and *establish conflict resolution plans for potential conflicts between UX and non-UX staff members*.

These practices are inspired by existing guidelines on how to improve the success of SPI efforts in general [79]. In creating these practices, we have updated and, when applicable, adjusted the recommended SPI practices to better suit UX integration. In addition, we have taken into consideration the implications of the five UX characteristics for stakeholder involvement to differentiate it from stakeholder involvement in SPI.

7 Discussion

The findings of the thesis shed light on various aspects of UX integration in general, and its challenges and success factors in particular. First, our findings show that UX integration challenges and success factors are diverse and concern both technical and organizational aspects of integration. However, the majority of these challenges and success factors concern the latter aspect which has not so far gained due attention in the software community (answers to RQ1 in Paper I and RQ5 in Paper IV). Second, these findings show that the five known unique characteristics of UX (*subjective*, *holistic*, *dynamic*, *context-dependent*, and *worthwhile*) have implications not only for the day-to-day work of practitioners but also for UX integration and its challenges and success factors. Nevertheless, the current research and industrial practice of integration often does not explicitly take these characteristics into account which in turn has contributed to a lopsided focus on the pragmatic aspect of UX (answers to RQ2 in Paper I and RQ3 in Paper II). Third, the findings show that, although not sufficiently acknowledged in the software community, UX integration is, in fact, a type of organizational change, and more specifically a type of Software Process Improvement (SPI) (answers to RQ4 in Paper III and RQ6 in Paper V and Paper VI). The findings of this thesis are summarized in Figure 3 in relation to both UX integration challenges (Paper I, Paper III, and

Paper IV) and success factors (Paper II-Paper VI) and also the organizational (Paper I-Paper VI) and technical aspects (Paper I, Paper II) of integration.

Based on the above findings, this thesis proposes various *UX integration principles and practices* to support practitioners in their integration efforts, and especially, to help them operationalize the insights gained from these findings. *UX principles and practices* and *UX integration principles and practices* are tightly connected, however, their main difference lies in their focus and purpose. While the former focuses on improving the UX delivered through the developed software, the latter focuses on improving the UX integration in organizations. The proposed UX integration principles and practices mainly aim to support practitioners in addressing the organizational issues concerning integration. They also explicitly take the unique characteristics of UX into account and aim to help companies move beyond usability integration to UX integration and prevent a lopsided focus on the pragmatic aspect of UX. Some of the practices we propose are based on previously known SPI practices which we have customized to better suit the nature of UX and UX integration. The proposed principles and practices are diverse and cover topics such as stakeholder involvement, organizational culture, communication and collaboration, and business model and strategy. Examples of these principles are *UX integration is a type of organizational change* and *resistance to UX integration may be shown by both UX and non-UX practitioners*. The practices, for instance, include *investigate and reflect on events outside the organization that may indirectly influence UX integration*, *establish conflict resolution plans for potential conflicts between UX and non-UX staff members*, *promote the benefits of improving the UX delivered through the products and services among the staff*, and *ensure that management provides strong leadership and support for UX integration and UX practices*.

This thesis highlights that for software companies to successfully benefit from the technical advances in the field of UX, addressing the organizational aspects of integration is a must as they, evidently, play an important and inevitable role in enabling or prohibiting UX integration in software companies. Organizational aspects of UX integration concern how and in what structure and setting UX and non-UX practitioners work together in the context of an organization in contrast to what specific practices they perform, or what tools, methods, and processes they use in their work (i.e. the technical aspects). The organizational aspects, for instance, include business alignment of the integration efforts, organizational culture, communication and collaboration between UX and non-UX practitioners, roles and responsibilities, power struggles, resources, etc. The thesis, therefore, adds to the current body of knowledge where the focus, so far, has been mainly on the technical aspects of UX integration. Examples are a variety of studies that propose different tools and methods to design or evaluate UX (e.g. emocards [90] for gathering users' momentary emotions about the interaction and UX curve [91]). While the findings of such studies are invaluable in supporting UX principles and practices and facilitating the day-to-day work of practitioners in software companies. However, they cannot be effectively applied and integrated into software companies without taking the organizational aspects into account. The thesis, therefore, advocates that in addition to supporting practitioners with regards to the technical aspects, researchers should also support these practitioners in systematically addressing the organizational aspects of integration; for instance, by developing principles, practices, tools, and methods suitable for such purpose. The thesis, therefore, takes an initial step in this regard and proposes the aforementioned principles and practices and a retrospective method that helps practitioners to systematically reflect on their previous integration efforts (Paper V).

In addition, this thesis highlights that to achieve a sustainable and successful UX integration in software companies, practitioners need to differentiate between UX and other software quality characteristics, in general, and usability in particular. In other words, these practitioners need to explicitly take into account the five unique characteristics of UX in their integration efforts to ensure addressing the hedonic aspect of UX and prevent a lopsided focus on its pragmatic aspect. Current literature on UX integration often does not clearly separate UX and usability (e.g. [27, 92]), therefore, does not reflect on the implications of the differences between these two concepts. UX and usability integration share the same essence. Therefore, practitioners can learn from and to some extent apply the existing usability integration body of knowledge also in the context of UX integration. However, these practitioners need to acknowledge that the characteristics of UX require a wider understanding of the end user compared to usability. This thesis shows that these characteristics have at least 20 implications (i.e. additional difficulties) for the day-to-day work of practitioners, nine of which are unique to this thesis. These characteristics also have implications for the integration, for instance, they contribute to power struggles between UX and non-UX practitioners, their communication and collaboration, and can even have an impact on forming a culture for or against UX in an organization. Practitioners, therefore, need to take these characteristics and their implications into account not only in their daily UX practices but also in their integration efforts. Some of the thesis's proposed UX integration principles and practices aim to help practitioners to overcome the lopsided focus on the pragmatic aspect of UX and to move beyond usability integration to UX integration. These practices and principles highlight that, among others, practitioners shall raise knowledge and awareness in different levels of the organization concerning: (i) how UX relates to other quality characteristics, especially usability, (ii) why both hedonic and pragmatic aspects of UX are important, (iii) what the implications of UX characteristics are for the day-to-day work of practitioners. Gaining such knowledge and awareness is important considering that the software community has a general lack of understanding regarding the relation of UX to other software quality characteristics as the findings of this thesis show. We hope to have been able to address this issue, at least to some extent, through our UX-aware model of requirements. The thesis also encourages other researchers to explicitly differentiate these two concepts and take their differences into account in their studies to provide research contributions that are more adjusted to the nature of UX and UX integration.

Additionally, the findings of this thesis underline the similarities between UX integration and organizational change, in general, and SPI in particular. This thesis shows that UX integration takes time, changes, and evolves over time. In addition, like other organizational changes, it can include a mixture of planned and emergent initiatives and can be influenced by events not only inside but also outside the organizations. Hence, this thesis extends the current research that often mainly focuses on planned and internal change initiatives to improve UX integration in organizations. For instance, the thesis argues that these organizations need to acknowledge that different decisions that are made outside the authority of UX practitioners can have an inevitable impact on enabling or prohibiting UX integration and, therefore, need to be identified and reflected on as part of the integration efforts. In other words, software companies need to not only control and influence the integration through planned and direct initiatives inside their organizations but also adjust them according to the presence and influence of emergent initiatives or external factors that may inevitably influence the integration. Therefore, this thesis argues that in their integration efforts, software companies can benefit from the organizational change and SPI body of knowledge. As an initial step, this thesis has customized a number of SPI practices to better suit UX integration efforts in software companies. As we mentioned before, the thesis also encourages the research community to broaden the horizon of UX integration research to also include the knowledge from the fields of organizational change and SPI. This broader perspective to integration can enable researchers to further support practitioners in their UX integration efforts by developing suitable tools and methods that address the real needs of these practitioners, in particular, with regards to the organizational aspects of integration and the hedonic aspect of UX.

We hope that the insights that this thesis provides about UX integration and its challenges and success factors together with the proposed UX integration principles and practices can facilitate and improve the current state of UX integration in the software industry. We hope to have been able to extend not only practitioners' but also researchers' understanding of UX integration, in particular, in relation to the hedonic aspect of UX, the organizational issues impacting the integration efforts, and the organizational change and SPI nature of UX integration. We hope that software companies can put this understanding into action by applying the thesis's proposed *UX integration principles and practices* to ensure moving beyond usability integration to UX integration and to prevent a lopsided focus on the pragmatic aspect of UX in their organizations. In addition, we hope that this understanding can further motivate researchers to explicitly differentiate UX and other quality characteristics (in particular usability) and address its unique characteristics in their research. We also hope that these researchers are more motivated to expand the horizon of their research towards other (often more mature) fields that can inspire UX integration research.

8 Threats to Validity

Threats to validity are outlined and discussed based on the classification by Runeson and Höst [73].

Construct validity concerns the alignment between what a researcher aims to measure and what actually is being measured. The selection process of subjects for interviews or workshops can cause a threat to construct validity in interview studies. Selection bias is always present when subjects are not

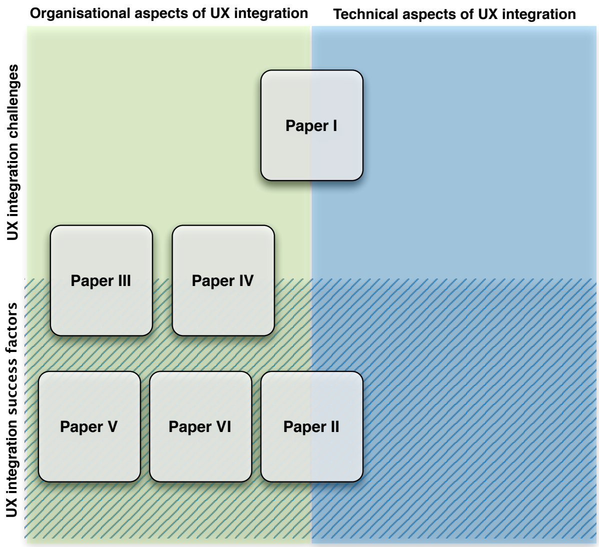
Organisational aspects of UX integration
Technical aspects of UX integration
UX integration challenges
UX integration success factors
Paper I
Paper III
Paper IV
Paper V
Paper VI
Paper II

Figure 3: This image shows how the contributions of the studies reported in the attached papers relate to (i) challenges and success factors of UX integration, (ii) technical and organizational aspects of UX integration.

fully randomly sampled. However, in our studies, the subjects were selected based on their role, experience, and availability so there is little more we could do to alleviate this threat. In addition, the presence of a researcher may influence the behavior and response of the subjects. This threat was alleviated somewhat by the guarantee of confidentiality of the data but is an inherent limitation of the research methods used in this thesis.

Internal validity concerns the causal relations between various phenomenon. In any empirical study, incorrect data is a threat to internal validity. In our studies, the interviews and workshops were audio recorded to mitigate this threat. We also analyzed the material in several rounds of independent as well as joint sessions to gradually reach consensus on the intended meaning of the respondents. The results of our analysis were also shared with the participants to validate and confirm the findings.

External validity concerns the generalizability of the findings and the extent to which the findings can be relevant to other contexts than the studied context. In our first study (Paper I), we sampled a number of different organizations in different industrial domains to decrease the effect of this threat. In addition, the majority of the organizations we studied are Swedish (exceptions are G (American), and F (Dutch)), and the culture of Swedish software industry can have an impact on how the studied organizations perceive and address UX or other quality characteristics. This distribution, however, is not sufficient to draw any conclusions based on the cultural differences of these companies, and in the interpretation of findings, this matter needs to be taken into consideration. The case company, represented in Paper III and Paper IV, is a medium-sized Swedish software development company, developing a business-to-business product. We, therefore, expect our findings to be valid for companies with similar characteristics. Qualitative studies rarely attempt to generalize beyond the actual setting and are more concerned with explaining and understanding the phenomena under study.

Reliability concerns the extent to which the data and analysis are dependent on the specific researchers. Although the coding process in the studies reported in Paper I, Paper IV, and Paper V performed by the first author, to improve the reliability of the generated themes, the three authors individually and independently conducted a pilot coding of these segments using an initial coding guide as explained above. The outcomes of the pilot coding were discussed in several sessions with all three authors, and the differences in coding were analyzed and resolved. Also, we had carefully designed the interviews before running them. We also defined the coding process after the interviews and before analyzing the data. The initial codes were therefore identified mainly based on observed interview responses. We also ensured the themes are not imposed on the data rather emerged from it.

9 Conclusions and Future Work

This thesis aims to facilitate and improve the current state of UX integration in the software industry. In doing so, the thesis presents new empirical findings on UX integration and its challenges and success factors. Based on these findings, the thesis then suggests principles and practices that can help practitioners in their integration efforts. The thesis argues that software practitioners need to ensure addressing the organizational aspects of integration and acknowledge that UX integration is a type of organizational change and more specifically, a type of software process improvement. In addition, these practitioners need to differentiate between UX and other software quality characteristics, in general, and usability in particular, hence, explicitly take into account the five unique characteristics of UX in their integration efforts in order to ensure addressing not only the pragmatic aspect of UX but also its hedonic aspect.

The studies included in this paper mainly contribute to the organizational aspects of UX integration, however, with a focus on UX characteristics. Although this was not from the beginning the focus of the thesis, the work moved in that direction as we gathered more data. The thesis also includes a UX-aware model of requirements and proposes a retrospective method that practitioners can apply to reflect on their UX integration initiatives. Through a better understanding of UX integration and its challenges and success factors, practitioners can systematically plan to overcome these challenges. Researchers can also use this knowledge to define and perform more industry-relevant research studies and provide practitioners with recommendations on how to address these challenges or benefit from the success factors in different organizational contexts. Although our focus has been on UX integration, the insights that can be gained from our findings may also shed light on integrating other multi-disciplinary concepts into the complex nature of organizations.

The work of this thesis can be extended by future research to further validate our findings (e.g. the proposed practices) and to also provide empirical data on aspects that our studies did not cover. As we mentioned, our studies did not aim to discover the cause and effect relationships between the identified challenges and success factors and an in-depth study is needed to identify and report on such relationships. Future research can also study how different organizational cultures or the type of software being developed (e.g., leisure vs work, business-to-business vs business-to-consumer) may impact the identified challenges or success factors in the studied organizations. Understanding this correlation can help organizations better identify and consider the relevance of the challenges and success factors to the characteristics of their organization and software. Such research not only helps to provide empirical support for the usefulness and effectiveness of the proposed UX integration principles and practices but also to refine them to better suit the needs of software development companies. Future research can also complement these proposed practices with a set of suitable tools and methods. Regarding our requirement model, the future search can also focus on providing even more detailed advice and examples on how to elicit, document, and break down UX requirements and refine them to other more concrete requirement types.

Bibliography

[1] M. Hassenzahl, "The Thing and I: Understanding The Relationship Between User and Product," in *Funology: from usability to enjoyment*, M. A. Blythe, A. F. Monk, K. Overbeeke, and P. C. Wright, Eds. Dordrecht: Springer, 2003, ch. 3, pp. 31–42.

[2] ISO, *ISO 25010: Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models.* Geneva, Switzerland: International Organisation for Standardisation, 2011.

[3] L. Chung, B. A. Nixon, E. Yu, and J. Mylopoulos, *Non-functional requirements in software engineering.* Boston, MA: Kluwer Academic Publishing, 2000.

[4] M. Hassenzahl and N. Tractinsky, "User experience - a research agenda," *Behaviour & Information Technology*, vol. 25, no. 2, pp. 91–97, mar 2006.

[5] ISO, *ISO 9241: Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems.* Geneva, Switzerland: International Organisation for Standardisation, 2010.

[6] J. Forlizzi and K. Battarbee, "Understanding experience in interactive systems," in *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques.* New York, NY: ACM, 2004, pp. 261–268.

[7] K. Väänänen-Vainio-Mattila, V. Roto, and M. Hassenzahl, "Now Let's Do It in Practice: User Experience Evaluation Methods in Product Development," in *Proceedings of Extended Abstracts on Human Factors in Computing Systems (CHI'08).* New York, NY: ACM, 2008, pp. 3961–3964.

[8] M. Hassenzahl, *Experience Design: Technology for All the Right Reasons.* San Francisco, CA: Morgan & Claypool, 2010.

[9] P. Wright and J. McCarthy, *Experience-Centered Design: Designers, Users, and Communities in Dialogue*, ser. Synthesis Lectures on Human-Centered Informatics. San Francisco, CA: Morgan & Claypool, 2010.

[10] M. Hassenzahl, "User experience (UX): towards an experiential perspective on product quality," in *IHM '08: Proceedings of the 20th International Conference of the Association Francophone d'Interaction Homme-Machine.* New York, NY: ACM, 2008, pp. 11–15.

[11] C. Lallemand, G. Gronier, and V. Koenig, “User experience: A concept without consensus? Exploring practitioners’ perspectives through an international survey,” *Computers in Human Behavior*, vol. 43, pp. 35–48, feb 2015.

[12] S. C. Chapra and R. P. Canale, *Numerical methods for engineers.* New York: Mcgraw-hill, 1998, vol. 2.

[13] P. Bourque and R. Fairley, eds, Eds., *Guide to the Software Engineering Body of Knowledge.* IEEE Computer Society, 2014.

[14] S. Abrahão, N. Juristo, E. L.-C. Law, and J. Stage, “Interplay between usability and software development,” *Journal of Systems and Software*, vol. 83, no. 11, pp. 2015–2018, nov 2010.

[15] J. Ferreira, H. Sharp, and H. Robinson, “Agile Development and User Experience Design Integration as an Ongoing Achievement in Practice,” in *Proceedings of the 2012 Agile Conference (AGILE 2012).* IEEE, 2012, pp. 11–20.

[16] T. Ovad and L. B. Larsen, “The Prevalence of UX Design in Agile Development Processes in Industry,” in *Proceedings of the 2015 Agile Conference (AGILE 2015).* IEEE, 2015, pp. 40–49.

[17] M. Isomursu, A. Sirotkin, P. Voltti, and M. Halonen, “User Experience Design Goes Agile in Lean Transformation – A Case Study,” in *Proceedings of the 2012 Agile Conference (AGILE 2012).* IEEE, 2012, pp. 1–10.

[18] J. Ferreira, H. Sharp, and H. Robinson, “User experience design and agile development: managing cooperation through articulation work,” *Software: Practice and Experience*, vol. 41, no. 9, pp. 963–974, 2011.

[19] B. Nuseibeh and S. Easterbrook, “Requirements engineering: A Roadmap,” in *Proceedings of the conference on The future of Software engineering - ICSE ’00*, vol. 1. New York, New York, USA: ACM Press, 2000, pp. 35–46.

[20] R. Berntsson Svensson, T. Gorschek, B. Regnell, R. Torkar, A. Shahrokni, and R. Feldt, “Quality Requirements in Industrial Practice- An Extended Interview Study at Eleven Companies,” *IEEE Transactions on Software Engineering*, vol. 38, no. 4, pp. 923–935, 2012.

[21] B. Paech and D. Kerlow, “Non-Functional Requirements Engineering - Quality is essential,” in *Proceedings of the 10th International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ’04)*, 2004, pp. 237–250.

[22] C. Ebert, “Putting requirement management into praxis: dealing with nonfunctional requiements,” *Information and Software Technology*, vol. 40, no. 3, pp. 175–185, jan 1998.

[23] S. Türpe and A. Poller, “Managing Security Work in Scrum: Tensions and Challenges,” *Proceedings of the International Workshop on Security in DevOps and Agile Secure Software Engineering (SecSE 2017)*, no. SecSE, pp. 1–16, 2017.

[24] G. Boström, J. Wäyrynen, M. Bodén, K. Beznosov, and P. Kruchten, "Extending XP Practices to Support Security Requirements Engineering," in *Proceedings of the 2006 International Workshop on Software Engineering for Secure Systems*, ser. SESS '06. New York, NY, USA: ACM, 2006, pp. 11–18.

[25] L. b. Othmane, P. Angin, H. Weffers, and B. Bhargava, "Extending the Agile Development Process to Develop Acceptably Secure Software," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 6, pp. 497–509, nov 2014.

[26] S. Rosenbaum, J. A. Rohn, and J. Humburg, "A toolkit for strategic usability," in *Proceedings of the SIGCHI conference on Human factors in Computing Systems (CHI'00)*, vol. 2, no. 1. New York, NY: ACM, 2000, pp. 337–344.

[27] D. Mostafa, "Maturity Models in the Context of Integrating Agile Development Processes and User Centred Design," PhD, University of York, 2013.

[28] C. Ardito, P. Buono, D. Caivano, M. F. Costabile, R. Lanzilotti, and Y. Dittrich, "Human-centered design in industry: Lessons from the trenches," *Computer*, vol. 47, no. 12, pp. 86–89, 2014.

[29] G. Venturi, J. Troost, and T. Jokela, "People, Organizations, and Processes: An Inquiry into the Adoption of User-Centered Design in Industry," *International Journal of Human-Computer Interaction*, vol. 21, no. 2, pp. 219–238, 2006.

[30] A. P. O. S. Vermeeren, E. L.-C. Law, V. Roto, M. Obrist, J. Hoonhout, and K. Väänänen-Vainio-Mattila, "User experience evaluation methods: Current State and Development needs," in *Proceedings of the 6th Nordic Conference on Human-Computer Interaction Extending Boundaries (NordiCHI '10)*. New York, NY: ACM, 2010, pp. 521–530.

[31] E. L.-C. Law, P. van Schaik, and V. Roto, "Attitudes towards user experience (UX) measurement," *International Journal of Human-Computer Studies*, vol. 72, no. 6, pp. 526–541, 2014.

[32] C. Gerea and V. Herskovic, "Measuring User Experience in Latin America: An Exploratory Survey," in *Proceedings of the Latin American Conference on Human Computer Interaction - CLIHC '15*. New York, NY: ACM Press, 2007, pp. 1–4.

[33] J. A. Rohn, D. Wixon, D. Rosenberg, J. Ashley, and L. Tesler, "Invited Panel: Managing UX Teams : Insights from Executive Leaders," in *Proceedings of Extended Abstracts on Human Factors in Computing Systems (CHI'12)*. New York, NY: ACM, 2012, pp. 1143–1146.

[34] R. Alves, P. Valente, and N. J. Nunes, "The state of user experience evaluation practice," in *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational (NordiCHI '14)*. New York, NY: ACM, 2014, pp. 93–102.

[35] K. Kuusinen and K. Väänänen-Vainio-Mattila, "How to make agile UX work more efficient: management and sales perspectives," in *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design (NordiCHI '12)*. New York, NY: ACM, 2012, pp. 139–148.

[36] L. Chung and J. C. S. do Prado Leite, "On Non-Functional Requirements in Software Engineering," in *Proceedings of Conceptual Modeling: Foundations and Applications*. Berlin, Heidelberg: Springer, 2009, pp. 363–379.

[37] T. T. Hewett, R. Baecker, S. Card, T. Carey, J. Gasen, M. Mantei, G. Perlman, G. Strong, and W. Verplank, "ACM SIGCHI Curricula for Human-Computer Interaction," Tech. Rep., 1992.

[38] M. Maguire, "Methods to support human-centred design," *International Journal of Human-Computer Studies*, vol. 55, no. 4, pp. 587–634, oct 2001.

[39] P. W. Jordan, *Designing Pleasurable Products: An Introduction to the New Human Factors*. London, New York: Taylor & Francis, 2002.

[40] B. J. Pine and J. H. Gilmore, "Welcome to the experience economy," *Harvard business review*, vol. 76, 1998.

[41] D. Rosenberg, "Introducing the business of UX," *interactions*, vol. 21, no. 1, pp. 74–76, jan 2014.

[42] R. Hartson and P. Pyla, *The UX Book: Process and Guidelines for Ensuring a Quality User Experience*. Morgan Kaufmann, 2012.

[43] L. Chapman and S. Plewes, "A UX Maturity Model : Effective Introduction of UX into Organizations," in *Design, User Experience, and Usability. User Experience Design Practice*. Springer, 2014, pp. 12–22.

[44] J. Choi, B.-J. Kim, and S. Yoon, "UX and Strategic Management: A Case Study of Smartphone (Apple vs. Samsung) and Search Engine (Google vs. Naver) Industry," in *HCI in Business: First International Conference, HCIB 2014*. Heraklion, Crete, Greece: Springer, 2014, pp. 703–710.

[45] B. May, "Applying Lean Startup: An Experience Report," in *Proceedings of the 2012 Agile Conference (AGILE 2012)*. IEEE, aug 2012, pp. 141–147.

[46] J. A. Rohn, "How to organizationally embed UX in your company," *Interactions*, vol. 14, no. 3, p. 25, may 2007.

[47] S. Mahlke, "User Experience of Interaction with Technical Systems: Theories, methods, empirical results and their application to the design and evaluation of interactive systems," PhD Dissertation, Berlin Technical university, 2007.

[48] J. McCarthy and P. Wright, "Technology As Experience," *Interactions*, vol. 11, no. 5, pp. 42–43, 2004.

[49] P. M. A. Desmet and P. Hekkert, "The basis of product emotions," in *Pleasure With Products: Beyond Usability*, W. S. Green and P. W. Jordan, Eds. London: Taylor & Francis, 2002, ch. 4, pp. 60–68.

[50] P. G. Zimmermann, "Beyond Usability-Measuring Aspects of User Experience," PhD Dissertation, Swiss Federal Institute of Technology Zurich, 2008.

[51] M. Thuring and S. Mahlke, "Usability, aesthetics and emotions in human-technology interaction," *International Journal of Psychology*, vol. 42, no. 4, pp. 253–264, 2007.

[52] P. Wright, J. McCarthy, and L. Meekison, "Making Sense of Experience," in *Funology*, ser. Human-Computer Interaction Series, M. Blythe, K. Overbeeke, A. Monk, and P. Wright, Eds. Springer Netherlands, 2005, vol. 3, ch. 4, pp. 43–53.

[53] M. Hassenzahl, R. Wessler, and K.-C. Hamborg, "Exploring and understanding product qualities that users desire," *Proceedings of the 5th Annual Conference of the Human-Computer Interaction Group of the British Computer Society (IHM-HCI)*, pp. 95–96, 2001.

[54] B. Martin and B. Hanington, *Universal Methods of Design: 100 Ways to Research Complex Problems, Develop Innovative Ideas, and Design Effective Solutions.* Rockport Publishers, 2012.

[55] K. M. Sheldon, a. J. Elliot, Y. Kim, and T. Kasser, "What is satisfying about satisfying events? Testing 10 candidate psychological needs." *Journal of personality and social psychology*, vol. 80, no. 2, pp. 325–39, mar 2001.

[56] M. Buchenau and J. F. Suri, "Experience prototyping," in *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques.* ACM, 2000, pp. 424–433.

[57] A. F. Newell, A. Carmichael, M. Morgan, and A. Dickinson, "The use of theatre in requirements gathering and usability studies," *Interacting with computers*, vol. 18, no. 5, pp. 996–1011, 2006.

[58] I. DIS, "9241-210: 2010. Ergonomics of human system interaction-Part 210: Human-centred design for interactive systems," *International Standardization Organization (ISO). Switzerland*, 2009.

[59] J. Gulliksen, G. Bengt, I. Boivie, J. Persson, S. Blomkvist, B. Göransson, and Å. Cajander, "Key principles for user-centred systems design," *Behaviour & Information Technology*, vol. 22, no. 6, pp. 397–409, nov 2003.

[60] ISO, *ISO/TR 18529 Ergonomics- Ergonomics of human- system interaction- Human-centred lifecycle process descriptions.* Geneva, Switzerland: International Organisation for Standardisation, 2000.

[61] J.-Y. Mao, K. Vredenburg, P. W. Smith, and T. Carey, "The state of user-centered design practice," *A Survey of User-Experience Development at Enterprise Software Companies*, vol. 48, no. 3, pp. 105–109, 2005.

[62] L. Karlsson, Å. G. Dahlstedt, B. Regnell, J. Natt och Dag, and A. Persson, "Requirements engineering challenges in market-driven software development - An interview study with practitioners," *Information and Software Technology*, vol. 49, no. 6, pp. 588–604, jun 2007.

[63] A. Borg, A. Yong, P. Carlshamre, and K. Sandahl, "The Bad Conscience of Requirements Engineering : An Investigation in Real-World Treatment of Non-Functional Requirements," in *Proceedings of the 3rd Conference on Software Engineering Research and Practice in Sweden (SERPS'03)*, 2003, pp. 1–8.

[64] J. O. Bak, K. Nguyen, P. Risgaard, and J. Stage, "Obstacles to Usability Evaluation in Practice," in *5th Nordic conference on Human-computer interaction building bridges - NordiCHI '08*. New York, New York, USA: ACM Press, 2008, p. 23.

[65] J. Gulliksen, I. Boivie, J. Persson, A. Hektor, and L. Herulf, "Making a difference: a survey of the usability profession in Sweden," in *Proceedings of the 3rd Nordic conference on Human-computer interaction (NordiCHI '04)*. New York, NY: ACM, 2004, pp. 207–215.

[66] Å. Cajander, M. K. Lárusdóttir, and J. Gulliksen, "Existing but Not Explicit - The User Perspective in Scrum Projects in Practice," in *Proceedings of the 14th IFIP TC 13 International Conference Human-Computer Interaction (INTERACT 2013)*. Berlin, Heidelberg: Springer, 2013, pp. 762–779.

[67] I. Boivie, J. Gulliksen, and B. Göransson, "The lonesome cowboy: A study of the usability designer role in systems development," *Interacting with Computers*, vol. 18, no. 4, pp. 601–634, jul 2006.

[68] S. Chamberlain, H. Sharp, and N. Maiden, "Towards a Framework for Integrating Agile Development and User-Centred Design," in *Proceedings of Extreme Programming and Agile Processes in Software Engineering (XP 2006)*. Berlin, Heidelberg: Springer, 2006, pp. 143–153.

[69] C. Lallemand, V. Koenig, and G. Gronier, "How Relevant is an Expert Evaluation of User Experience based on a Psychological Needs-Driven Approach?" in *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational (NordiCHI '14)*. New York, NY: ACM, 2014, pp. 11–20.

[70] M. Hassenzahl, M. Burmester, and F. Koller, "AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität (Attraktif: A questionnaire for the measurement of perceived hedonic and pragmatic quality)," in *Mensch & Computer 2003: Interaktion in Bewegung*, J. Ziegler and G. Szwillus, Eds. Stuttgart: B. G. Teubner, 2003, pp. 187–196.

[71] M. Hassenzahl, S. Diefenbach, and A. Göritz, "Needs, affect, and interactive products- Facets of user experience," *Interacting with Computers*, vol. 22, no. 5, pp. 353–362, sep 2010.

[72] C. Robson, *Real World Research: A Resource for Users of Social Research Methods in Applied Sett.* Wiley, 2011.

[73] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Software Engineering*, vol. 14, no. 2, pp. 131–164, 2008.

[74] P. Runeson, M. Host, A. Rainer, and B. Regnell, *Case Study Research in Software Engineering: Guidelines and Examples*, 1st ed. Wiley Publishing, 2012.

[75] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative Research in Psychology*, vol. 3, no. 2, pp. 77–101, 2006.

[76] A. L. Strauss and J. Corbin, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*, 2nd ed. Sage Publications, 1998.

[77] B. G. Glaser, A. L. Strauss, and E. Strutzel, "The Discovery of Grounded Theory; Strategies for Qualitative Research." *Nursing Research*, vol. 17, no. 4, p. 364, 1968.

[78] K.-j. Stol, P. Ralph, and B. Fitzgerald, "Grounded theory in software engineering research," in *Proceedings of the 38th International Conference on Software Engineering - ICSE '16*. New York, New York, USA: ACM Press, 2016, pp. 120–131.

[79] M. Niazi, D. Wilson, and D. Zowghi, "A maturity model for the implementation of software process improvement: An empirical study," *Journal of Systems and Software*, vol. 74, no. 2, pp. 155–172, 2005.

[80] B. Burnes, "Emergent change and planned change – competitors or allies?: The case of XYZ construction," *International Journal of Operations and Production Management*, vol. 24, no. 9, pp. 886–902, 2004.

[81] J. Heiskari, M. Kauppinen, M. Runonen, and T. Mannisto, "Bridging the Gap Between Usability and Requirements Engineering," in *Proceedings of the 17th IEEE International Requirements Engineering Conference*. IEEE, aug 2009, pp. 303–308.

[82] T. Gilb and L. Brodie, *Competitive Engineering: A Handbook for Systems Engineering Requirements Engineering, and Software Engineering Using Planguage.* Elsevier Ltd., 2005.

[83] R. Berntsson Svensson and B. Regnell, "A Case Study Evaluation of the Guideline-Supported QUPER Model for Elicitation of Quality Requirements," in *Proceedings of the 21st International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ'15)*, 2015.

[84] T. Gorschek, C. Wohlin, P. Carre, and S. Larsson, "A Model for Technology Transfer in Practice," *IEEE Software*, vol. 23, no. 6, pp. 88–95, 2006.

[85] E. Eriksson, J. Gulliksen, and Å. Cajander, "Introducing usability roles in public authorities," *Proceedings of the 5th Nordic conference on Human-computer interaction building bridges - NordiCHI '08*, p. 113, 2008.

[86] J. Winter and K. Rönkkö, "SPI success factors within product usability evaluation," *Journal of Systems and Software*, vol. 83, no. 11, pp. 2059–2072, 2010.

[87] C. Raison and S. Schmidt, "Keeping user centred design (UCD) alive and well in your organisation: Taking an agile approach," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8012 LNCS, no. PART 1, pp. 573–582, 2013.

[88] E. Bjarnason, A. Hess, R. Berntsson Svensson, B. Regnell, and J. Doerr, "Reflecting on Evidence- Based Timelines," *IEEE Software*, vol. 31, no. 4, pp. 37–43, 2014.

[89] P. Kashfi, R. Feldt, A. Nilsson, and R. B. Svensson, "Evidence-based Timelines for User eXperience Software Process Improvement Retrospectives," in *42nd Euromicro Conference on Software Engineering and Advanced Applications.* IEEE, aug 2016.

[90] P. Desmet, K. Overbeeke, and S. Tax, "Designing Products with Added Emotional Value: Development and Application of an Approach for Research through Design," *The Design Journal*, vol. 4, no. 1, pp. 32–47, 2001.

[91] S. Kujala, V. Roto, K. Väänänen-Vainio-Mattila, and A. Sinnelä, "Identifying hedonic factors in long-term user experience," in *Proceedings of the 2011 Conference on Designing Pleasurable Products and Interfaces (DPPI '11).* New York, NY: ACM, 2011, pp. 17:1–17:8.

[92] L. Plonka, H. Sharp, P. Gregory, and K. Taylor, "UX Design in Agile: A DSDM Case Study," in *Proceedings of the 15th International Conference on Agile Processes in Software Engineering and Extreme Programming (XP 2014).* Springer International, 2014, pp. 1–15.