

Software development methods and usability: Perspectives from a survey in the software industry in Norway

Bendik Bygstad ^{a,*}, Gheorghita Ghinea ^{a,b}, Eivind Brevik ^a

^a Norwegian School of Information Technology, Schweigaards gt 14, 0185 Oslo, Norway

^b School of Information Systems, Computing and Mathematics, Brunel University, UB8 3PH, UK

Received 24 October 2006; received in revised form 17 November 2007; accepted 19 December 2007

Available online 10 January 2008

Abstract

This paper investigates the relationship between software development methodologies and usability. The point of departure is the assumption that two important disciplines in software development, one of software development methods (SDMs) and one of usability work, are not integrated in industrial software projects.

Building on previous research we investigate two questions; (1) Will software companies generally acknowledge the importance of usability, but not prioritise it in industrial projects? and (2) To what degree are software development methods and usability perceived by practitioners as being integrated? To this end a survey in the Norwegian IT industry was conducted. From a sample of 259 companies we received responses from 78 companies.

In response to our first research question, our findings show that although there is a positive bias towards usability, the importance of usability testing is perceived to be much less than that of usability requirements. Given the strong time and cost pressures associated with the software industry, we believe that these results highlight that there is a gap between intention and reality. Regarding our second research question our survey revealed that companies perceive usability and software development methods to be integrated. This is in contrast to earlier research, which, somewhat pessimistically, has argued for the existence of two different cultures, one of software development and one of usability. The findings give hope for the future, in particular because the general use of system development methods are pragmatic and adaptable.

© 2008 Elsevier B.V. All rights reserved.

Keywords: Software development methods; Usability; Software industry; Survey

1. Introduction

This paper investigates the relationship between two important disciplines of modern software development; the use of software development methods and the concepts and techniques of usability.

Software development methods (SDM) have been in use for the past forty years and constitute a core part of modern software engineering (Sommerville, 2001; Jacobson et al., 1999). Still, they represent a thorny issue, both because their effectiveness has been challenged (Fitzgerald, 1998; Truex et al., 2000) and because of the continuous

wars between proponents of different methods (Larman and Basili, 2003). Previous research has also shown that most successful companies do not use such methods mechanistically; rather they pick and choose elements that are congruent with existing work practices (Fitzgerald, 1998; Bygstad et al., 2004). During the 1990s most new methods introduced were iterative and incremental, acknowledging the instability of initial user requirements. Well-known examples are Rational Unified Process (Jacobson et al., 1999), DSDM (Stapleton, 2003), Microsoft Solutions Framework (Microsoft, 2004) and XP (Beck, 2000).

Usability, on the other hand, emerged during the mid 1980s, and was embraced in the 1990s by the software industry. This was partly as a response to the new challenges that web based software – to be used by a large

* Corresponding author.

E-mail address: bendik.bygstad@nith.no (B. Bygstad).

number of diverse users – put on developers. The body of knowledge of usability is large and includes various perspectives, from usability engineering (Nielsen, 1993) to more context-oriented approaches (Beyer and Holzblatt, 1997).

This paper investigates empirically, through a survey among Norwegian IT companies, the relationship between SDM and usability in current industry practice. We investigate which SDMs that are adopted, and to which degree the companies have adopted usability techniques. These findings are used to investigate our core assumption – that SDMs and usability work (Iivari, 2006) are both accepted as best practices in principle, but not yet integrated in a full process.

The paper is structured as follows. In Section 2, we discuss findings in earlier research, and present our two research questions. Then, in Section 3, we briefly present our research method. In the Section 4, the result of research will be presented followed by a discussion. Section 5 concludes and points to further research.

1.1. Research review: software development and usability

Usability is a key property of interactive systems. Testimony to this lie the standardization efforts associated with usability (see, for instance, ISO, 1998, 1999) as well as the considerable body of research in the area. Although few would argue against making interactive systems and software products ‘usable’, actually doing so in practice has proven to be a challenging endeavour, and considerable efforts have been undertaken to identify the obstacles to integrating usability issues in software development (Boivie et al., 2003; Grudin, 1993; Gunther et al., 2001; Kujala, 2003; Poltrock and Grudin, 1994; Rosenbaum et al., 2000; Vredenburg et al., 2002).

Although SDMs and usability work have some similarities (they are both applied disciplines, and they play important roles in software development), their differences are much more obvious. While SDMs originated from systems engineering and software economics (Sommerville, 2001) in the late 1960s, usability work was developed in the late 1980s and early 1990s from HCI research, cognitive psychology and ergonomics. While software development was – with some notable exceptions – mainly concerned about the inner workings of the system, usability focused on the user. Thus, the role of the user is different; in software development the user is primary a means to elicit requirements (Jacobson et al., 1999), while for usability work the users are the prime means for designing the system (Gould and Lewis, 1985; Karat, 1997; Nielsen, 1993; Ehn and Löwgren, 1997). This difference in perspectives does not imply that users are unimportant in SDMs; rather it indicates that, in the immensely complicated task of constructing high quality software systems, usability of one of many challenges.

A central issue that has arisen in this respect is that, while designing for usability mandates user involvement

(ISO, 1999) in the software development cycle, there is no clear process definition of how this should happen. For instance, it is generally accepted that the principles espoused by Gould and Lewis (1985), namely:

1. An early focus on users and tasks.
2. Empirical measurement.
3. Iterative design.

as well as those subsequently put forward by other researchers (Gulliksen et al., 2003) do lead to the development of usable software products. Nonetheless, the extent to which these are followed depends on the particular development methodology adopted (Kujala, 2003).

Software development theorists tend to play down these differences, arguing that usability may easily be integrated into the formal frameworks. The iterative and incremental structure of modern SDMs ensures continuous communicational and validation by users (Jacobson et al., 1999; Stapleton, 2003). In his general text book on software engineering Sommerville (2001) includes a long chapter on User-Interface Design, stating that “good user interface design is critical to the success of a system” (p.328). The widely used Rational Unified Process focuses on both usability testing and on User-Centered Design, which are explained in these terms:

“Usability tasks should be performed in parallel early in development. These tasks would include sketching the user interface and drafting the user guides or online help. (John) Gould also makes the point that usability should be the responsibility of one group.

An important feature of integrated design is that the overall approach – the framework – for detailed user-interface design is developed and tested at an early stage. This is an important difference between user-centered design and other purely incremental techniques. It ensures that incremental design carried out in later phases fits seamlessly into the framework and that the user interface is consistent in appearance, terminology and concept” (Rational Corp, 2001).

Correspondingly, agile software engineering frameworks, such as XP (Beck, 2000) and DSDM (Stapleton, 2003) methods recommend very close interaction between business users and developers, and also include various forms of usability work. (Constantine, 2002; Larman, 2004).

Usability researchers tend not to accept these “extensions”, and have argued that the differences are much deeper, because traditional software engineering is basically focused on producing high quality *code*. These differences, it is contended, add up to two different *cultures* of software development (Boivie et al., 2006; Iivari, 2006; Jerome and Kazman, 2005), and have called for fundamentally new approaches to counter the basically technical approach of SDMs. For example, Boivie et al. (2006) concluded – after a review of this relationship – somewhat pessimistically:

“We believe that one of the main difficulties with incorporating User Centric Systems Development in existing processes is that it requires a great deal more than simply adding a few activities to existing processes. It requires new development approaches, new methods, new roles, new ways of planning and allocating resources etc. Moreover, a user-centered approach changes the relationship between the user/client organization and the development organization (..)” (Boivie et al., 2006).

Moreover, there is agreement amongst researchers that there are gaps between developers and users (Borgholm and Madsen, 1999; Grudin, 1991;), and consequently work has been undertaken to provide guidance for user involvement across roles and management layers of a project (Boivie et al., 2003; Clegg et al., 1997; Damodaran, 1996; Hutchings and Knox, 1995; Gulliksen et al., 2006; Iivari, 2006; Mrazek and Rafeld, 1992).

Notwithstanding these efforts, the usability designer is still perceived to be a singular advocate in organizations (Boivie et al., 2006) and recent research has led Iivari (2006) to conclude that:

“...there might not be one ‘best, universally valid, context free way’ of introducing and carrying out usability work in software product development organizations.”

The question of what, therefore, happens with usability considerations in practice, and what their relationship to software product development actually, is one that needs answering. As shown by Kujala (2003) it is difficult to analyse this academic debate, because the effectiveness of the different approaches are not documented. Most SDM and usability research are dominated by normative approaches, focusing on what should be done, not what actually happens in practice.

As Table 1 illustrates, previous work exploring usability issues in a ‘real-world’ context, has either concentrated on surveys of professionals (with sample sizes of varying magnitudes), or on case studies with a rather limited number of cases. With the notable exception of Rauch and Wilson’s (1995) survey of usability from a large cross-section of companies, to the best of our knowledge, a survey of companies with respect to this subject has not yet been undertaken.

Moreover, given that previous studies (Damodaran, 1996; Gulliksen et al., 2004) have indicated that management support is essential for including usability considerations in projects, it seems natural to ask managers about this point – which is again one of the issues that we address in this paper.

1.2. Research questions

Our point of departure, therefore, is that these issues should be investigated in an industrial context. From this discussion in the review we pose two research questions. The first is concerned with the general status of usability in software development.

RQ1: Will software companies acknowledge the importance of usability, but not prioritize it in industrial projects?

We intend to investigate whether there is a gap between intention and reality; that the companies will express concern for usability, but not be willing to use resources on it in industrial projects with strong time and cost pressures.

The second question is concerned with the perceived relationship between software development methods and usability. We assume that most companies use some kind of method and that they also relate to usability issues. There is, however, a need to investigate to which degree these are integrated in the practices of the development projects.

Table 1
A snapshot of previous usability surveys and case studies

Author(s)	Number of respondents	Number of organisations	Countries/region
Bekker and Vermeeren (1996)	23	Not specified	Netherlands
Boivie et al. (2003)	40	2	Sweden
Boivie et al. (2006)	Not specified	2	Sweden
Borgholm and Madsen (1999)	Not specified	6	Denmark, U.S.A.
Catarci et al. (2002)	Not specified	2	Italy
Clegg et al. (1997)	45	Not specified	U.K.
Gould and Lewis (1985)	447	Not specified	Not Specified
Grudin and Poltrock (1989)	200	7	Not specified
Gulliksen et al. (2004)	194	Not specified	Sweden
Gunther et al. (2001)	100		
Iivari (2006)	Not specified	5 Units across 3 organisations	Not specified
Jerome and Kazman (2005)	96	Not specified	Not specified
Poltrock and Grudin (1994)	Not specified	2	Not specified
Rosenbaum et al. (2000)	134	Not specified	Not specified
Vredenburg et al. (2002)	103	Not specified	USA (60%) and Europe
Rauch and Wilson (1995)	226	At least 119	Not specified

RQ2: To what degree are software development methods and usability perceived by practitioners as being integrated?

In the next section, we outline how the research questions were investigated.

2. Research method

This section will first give a description of the sampling and sampling design that has been used. Then research design and analysis of survey responses are determined.

2.1. Sampling and sampling design

The greatest sampling challenge in this type of research is to identify which companies that actually engage in software development (Fitzgerald, 1998). This study builds on similar studies done annually in Norway between 2002 and 2004 (Bygstad et al., 2004), where a great effort was put into establishing a population of Norwegian IT companies that engage in software development. Ideally, all companies involved in software development in Norway should be defined as the population for this research. This includes general private companies and public organisations as well as professional companies within the IT sector.

The earlier studies showed, however, that response rates from general private and public companies were too low to be useful. Thus, the population was limited to IT companies in the following three different Norwegian industrial classification (IC) codes:

7220000 System – and software consulting
7260001 IT consulting
7260003 IT services

In keeping with our perspective expressed in the research review, the targeted respondents were the general or IT managers in these companies. Our sample was collected from two sources. First it consists of the 194 companies that accepted to participate in 2003. Second, this was supplemented by 65 companies that participate as partners in NITH student development projects, which we knew were engaged in software development. These companies belonged all to the same industrial groups as the first group. Of course, this sampling strategy puts some limitations on the implications of our findings, which we will return to in our discussion. For both the respondent groups, however, we had assured that the individual respondents were knowledgeable with respect to the systems development environment in their companies. The first group had been asked – and accepted – in a telephone request to participate in a detailed survey on software development. The second group consisted of managers we knew personally from the student projects as being in charge of the company's projects.

2.2. Research design

A questionnaire was designed, with 5 questions on SDMs and 8 questions on usability. We also asked how many persons were engaged in software development in the company. The respondents were also invited to comment on each of the topics of the survey. The questionnaire is documented in [Appendix A](#).

The survey was implemented electronically by using the QuestBack system.¹ This system is based on e-mail distribution of a link to the actual survey and replies via a web browser on the Internet. The QuestBack system has an automatic reminder, which was scheduled once to those who had not responded after the request to participate in the survey was sent out.

After a four weeks period in June 2006, the survey was closed with 78 responses, representing a response rate of 33%.

3. Results and discussion

This section presents the results and discussion, and is divided into three parts: (1) Adoption of SDM (2) Usability in requirements and testing and (3) The relationship between SDMs and usability. The first two sections are descriptive, while we examine our research questions in part 3.

3.1. Adoption of SDM

The respondents were asked whether or not they were using a formal SDM.

As shown in [Table 2](#) the majority do not use a formal method, but a number of techniques and tools. The respondents that answered 'yes' were then asked to indicate which formal SDMs that were in use. The result is shown in [Table 3](#).

The sum of percentages is greater than 100% because some companies use more than one method. A large majority, 68%, of software development companies uses their own method. This is in line with the findings of the 2002 and 2003 surveys. The numbers do not provide evidence of a significant change in the usage of commercial methods. Rather, they suggest that companies tend to stick to a certain method, and are reluctant to change. The comments from the companies illustrate this point; they are generally quite satisfied with their choice of method. Some quotes:

“Our own method is adaptive.”

“We chose MSF, which had the necessary flexibility related to the continuous changes in requirements during the development projects.”

“We have very good experience with use case based development.”

¹ www.questback.com.

Table 2
Formal SDM use

Answer	N	Percent (%)
Yes	27	35
We do not use a formal SDM, but we use a number of techniques and tools	45	57
No	6	8
Total	78	100

Table 3
Formal SDM usage (multiple answers possible)

Method	2006 (%)	2003 (%)	2002 (%)
Own method	68	78	79
Rational unified process	29	29	23
XP/Agile methods	18	21	17
Microsoft Solutions Framework	29	19	21
Other methods	19	10	13

“We use a waterfall method, with our own checklists.”

“It is a bit bureaucratic to use the whole method, but we adapt it, and use small mini projects, doing early tests...”

In the next section, we present the results of our questions on usability.

3.2. Adoption of usability techniques

Designing for usability typically involves establishing user requirements for a new system, iterative design and testing with representative users (Gould and Lewis, 1985). Thus, in order to examine the interplay between usability and system development methods, in our survey we specifically sought to explore to which degree usability was included in the system requirements and the degree of usability testing. Usability in requirements was measured by two questions, the first being “*When will you include usability in requirements?*” The result is shown in Table 4.

Several respondents indicated views such as:

“Usability is an important competitive advantage.”

“It is not possible to sell our software without a high degree of usability.”

Table 4
Usability and requirements

Answer	N	Percent (%)
Always	55	72
Only if usability problems emerge during the project	8	10
Only if the customer demands it	12	15
Only if we have an internal usability specialist available	2	3
Total	77	100

“Usability is crucial for our customers at all levels.”

Another respondent, however, stated:

“I wish we did!”

Thus, although there are some exceptions, there is a general appreciation of the importance of usability in giving companies competitive advantage.

The second question on was “*How do you collect requirements for usability?*” Results are shown in Table 5.

The replies highlight that the majority of companies trust their previous experience in eliciting usability requirements, and that two thirds interview users as a part of this process.

The respondents were also asked two questions on usability testing. The first was “*How many users are typically engaged in usability testing?*” As Table 6 shows, the samples of users in testing are generally small, most being less than 10 users.

Table 7 shows how these users were selected. It shows that 40% of the respondents report that they select a representative sample of users. 23% of the respondents state that the test users were selected by the customer’s employees, and one may ask whether these users were selected repre-

Table 5
Usability and requirements (multiple answers possible)

Answer	N	Percent (%)
Interviewing users	52	67
Best practice from earlier projects	54	71
Books, Internet resources	15	19
Other	9	12

Table 6
Number of users in usability testing

Answer	N	Percent (%)
1–10 users	52	67
11–50 users	17	21
More than 50 users	2	3
We do not test usability	7	9
Total	78	100

Table 7
Selection criteria for users in usability testing

Answer	N	Percent (%)	How important is usability testing? (scale 1 to 6)
Arbitrary sample of users	4	5	3.2
Representative sample of users	31	40	4.6
Own employees	7	9	4.0
Customer’s employees	18	23	3.7
Other	11	14	4.0
Do not test usability	7	9	
Total	78	100	

sentatively, but our data suggests no. As illustrated in Table 7, building on the responses on question 13, these companies had a low score (3,7) on the importance of usability testing. Indeed, a *t*-test between the two cohorts reveals that the differences in opinions between the two groups (representative sample of users vs. customer’s employees) with respect to usability testing is statistically significant ($t = -2.535$; $p < 0.05$). Moreover, it is interesting to note that for both these cohorts, the majority uses a sample of between 1 and 10 users for their tests (21 out of the 31 companies who employ a representative sample of users, and 14 out of the 18 companies who use the customer’s employees in their testing, are in this category). Somewhat reassuringly, the only 2 companies in our survey who use more than 50 users in their usability tests indicated that these were representatively sampled.

Summarizing the findings on usability, the results show that the majority of the respondents include usability in their requirements, and that they also collect usability requirements by including users in the process (Tables 4 and 5). In usability testing, however, the number of users seems quite small, as most of the companies only include less than 10 users (Table 6). Whilst this is not necessarily a bad thing (Nielsen for instance advocates not involving more than 5 users in usability testing), what is more worrying is that only about 40% of the users selected for testing are a representative sample of the users.

3.3. The relationship between SDM and usability

Returning to our two research questions we first asked:

- *RQ1: Will software companies generally acknowledge the importance of usability, but not prioritize it in industrial projects?*

To investigate this question we first assess the answers of two general questions on usability. The respondents were asked – in general terms – how important usability requirements and usability testing was for the success of their projects (Questions 10 and 13, respectively). The results are shown in Table 8.

As Table 8 shows 92% of companies surveyed think usability is important for the success of their projects,

awarding ratings of 4 or above, when polled on this issue. Somewhat surprising, but in line with the participatory design tradition, usability requirements are considered more important than usability testing.

However, when assessing the answers of the more concrete questions on usability activities in projects, the results show a different picture. Concerning usability requirements, 72% of the companies always include it, and almost 67% also interview the users, as showed in Tables 4 and 5. On the other hand, only 40% of the companies use a representative sample of users for usability testing (Table 7). Further, the number of users engaged in usability testing is generally quite small, as showed in Table 6.

Indeed, correlation analysis between responses to Question 10 (vis a vis the importance of usability requirements for the success of companies’ projects) and Question 13 (vis a vis the importance of usability testing for the success of companies’ projects) reveal a statistically significant correlation, with a Spearman rho coefficient of $\rho = 0.575$ ($p < 0.001$), which shows that respondents who appreciated the importance of collecting usability requirements generally also appreciated the importance of usability testing for the success of their projects. However, as Table 8 shows, while 92% of respondents displayed a positive bias in respect of the importance of usability requirements, this figure drops to 68% as far as positive bias towards usability testing is concerned. Further analysis (via a pairwise *t*-test) shows that this difference in mean opinion scores is, indeed, significant ($t = -6.758$; $p < 0.001$). This point is further put into perspective, since our survey data also revealed that 3 out of the 25 companies who indicated that usability requirements are very important for the success of their project actually then went on to confess that they do not test usability.

Similarly, correlation analyses between responses to Question 10 and Question 14 (vis a vis the degree of integration of usability in companies’ SDM) of our survey revealed a statistically significant correlation ($\rho = 0.537$; $p < 0.001$), as did responses between Question 13 and 14 ($\rho = 0.631$; $p < 0.001$). However, as Table 8 shows, it is again to be remarked that in spite of these positive correlations, the percentage of respondents who were actually positive with respect to the integration of usability in their SDM again drops even further to 63%. It is somewhat

Table 8
Usability requirements, usability testing – and project success; degree of integration of usability in SDM

Possible answers (Q10 and Q13)	Usability requirements (Q10) (%)	Usability testing (Q13) (%)	Integration of usability in SDM (Q14) (%)	Possible answers (Q14) (%)
6 – Very important	33	14	14	6 – To a large degree
5	38	23	23	5
4	21	31	26	4
3	6	19	18	3
2	1	6	13	2
1 – Quite unimportant	1	5	3	1
0 No answer			3	0
Total	100	100	100	Total

unsurprising therefore that a pairwise *t*-test analysis reveals that the differences in mean opinion scores between Questions 10 and 14 are again statistically significant ($t = -6.425$; $p < 0.001$).

In concluding, we find that there is a gap between intention and reality: the companies express interest and concern for usability, but this stance is not corroborated by their subsequent responses, which reveal that they are less willing to use resources on it in industrial projects with strong time and cost pressures. Moreover, it confirms, from an industrial viewpoint, what other developer-focused studies (Gulliksen et al., 2006) have highlighted, namely that there is a certain “fuzziness” in the integration of usability issues in software development.

In keeping with Damodaran’s (1996) point about there being effective participation of users in development project, we can remark that, based on our results, one possible reason why companies do not prioritise usability issues is precisely that such considerations are not seen by companies as being ‘effective’ and, therefore, loosely included in the process.

Our second question was:

- *RQ2: To what degree are software development methods and usability perceived by practitioners as being integrated?*

The respondents were asked “*To which degree do you think that usability is integrated in your software development method (whether you use a formal SDM or not)?*” The result is shown below in Table 9.

The table shows a positive bias in this respect, as 63% of the respondents answer that usability to some degree (values 6 to 4) is integrated in the development method. Moreover, a χ^2 revealed that this bias is also statistically significant ($\chi^2 = 28.795$; $p = .000$).

An interesting question is whether the adopted SDM is associated with different answers on this question. The answer is no, since we find no significant associations when we relate these findings, via a chi square test, to the adopted SDM. It does not affect this profile whether the SDM is the company’s own or a commercial method, neither if the company uses a formal SDM nor only a set of techniques.

We interpret this result as an indication that SDMs are not perceived as barriers to usability work. Regardless of

the type of SDM, for the majority of practitioners usability work may be included in industrial software development projects. Thus, it is reasonable to assume that the attitude to usability in the SDM community has changed during the past decade. This is also strengthened by several respondent comments:

“We used to have an ‘engineering approach’ to development, i.e., the engineers develop the system, design the user interface as if everybody else is an engineer. We have now hired a usability specialist as a consequence of this problem.”

“We used to do internal [usability] testing of our system. We have now decided to hire an external company specialising in usability. They will test the software at the customers’ sites.”

These findings do not support the somewhat pessimistic view from several usability researchers (Boivie et al., 2006; Iivari, 2006); that the two cultures are irreconcilable. First, we have documented that most IT companies do not view formal SDM as rigid frameworks; rather they pick and use elements that integrate with their existing work practices. This situation makes it much easier to also integrate usability work, thus aiming for a more holistic approach of software development called for by Ehn and Löwgren (1997). Second, the IT companies in this survey do view usability as a key factor for project success. What may still be lacking is a clearer role for usability work, as also suggested by Boivie et al. (2006).

3.4. Limitations

We acknowledge that there are limitations to this research. The 259 companies do not represent a random sample of the IT company population. However, although our survey respondents represent a sizeable proportion of the Norwegian software development sector, we do not claim it is representative – indeed, this was not the focus of our work. Our sample was initially selected from SDM companies, and one may question whether usability is an issue for all of them, for example for companies developing technical middleware. However, only 2 respondents reply that usability is unimportant in requirements (Table 8), indicating that our sample is valid also for usability issues.

In hindsight, we acknowledge that the questionnaire could have been improved. This applies in particular to the *actual use* of the SDMs (not only adoption), and to usability in requirements, where a more general question on the requirements activities could have yielded more information on this issue. We could also have asked more specifically on the participation of HCI professionals in the projects and have targeted specific aspects pertaining to the main ISO standards in the area – ISO 9241-11 and ISO13407 – these form valuable avenues for future work.

Lastly, with regards to the questionnaire, one may question whether the respondents have the same understanding

Table 9
To which degree is usability integrated in software development method?

Answer	N	Percent (%)
6 – To a large degree	11	14
5	18	23
4	21	26
3	14	18
2	10	13
1 – Not at all	2	3
No answer	2	3
Total	78	100

of the usability terms as in the IS research community. Moreover, one may also question whether the survey explored actual usability/SDM issues, as opposed to perceptions of respondents regarding. However, we will argue that this is an issue not only for our questionnaire, but within HCI-research in general. Actually, Gray and Salzman state that construct validity “is the single most important issue facing HCI researchers and practitioners” (1998:22).

4. Conclusions and future work

This paper investigated the adoption of software development methods and usability, through a web based survey in the software industry in Norway. The significance of this research is that it addresses and extends earlier case study research on SDMs and usability, within an industrial perspective. The point of departure was the assumption that two important practices in software development, one of traditional software development methods and one of usability work, are not integrated in industrial software projects.

Our research questions were: (1) *Will software companies generally acknowledge the importance of usability, but not prioritise it in industrial projects?* and (2) *To what degree are software development methods and usability perceived by practitioners as being integrated?*

In response to our first research question, our findings show that although there is a positive bias towards usability, the importance of usability testing is perceived to be much less than that of usability requirements. Given the strong time and cost pressures associated with the software industry, we believe that these results highlight that there is a gap between intention and reality, and it would be interesting to explore further what are the driving forces behind this.

As regards our second research question, our survey revealed that companies perceive usability and software development methods to be integrated. This, we believe, is a positive development, as it is in contrast to earlier research, which has, in this respect, assumed the existence of two different cultures. Whether this finding actually represents a trend, is, again, a worthy avenue for future work.

Appendix 1. Questionnaire (with results in %)

The survey was implemented electronically by using the QuestBack system (www.questback.com), and the questions were in Norwegian.

Question (1)	Do you use a formal systems development method (SDM)? With ‘use’ we understand that one uses the main structure or principles of the method.	
	<i>Yes</i>	35%
	<i>No, we use techniques and tools, but no method</i>	57%
	<i>No</i>	7%
Question (2)	What SDM has been in use by your company the last two years? (It is possible to tick off more than one alternative)	
	<i>Own method (developed within the company)</i>	68%
	<i>Rational Unified Process (RUP)</i>	29%
	<i>Microsoft Solutions Framework (MSF)</i>	18%
	<i>Extreme Programming (XP) or other agile methods</i>	29%
	<i>Other, please specify below</i>	19%
Question (3)	Describe in short what experiences you have with use of this/these methods listed in question 2)	
Question (4)	Choose the type of project that is most common in your company	
	<i>Internal development projects</i>	47%
	<i>Development projects for a client</i>	67%
	<i>Adaptation of commercial software</i>	18%
	<i>Other, please specify below</i>	13%
Question (5)	Do your company collect information about the degree of success in completed IT projects?	
	<i>Yes, internally in our own development department</i>	32%
	<i>Yes, from the management of the client/customer</i>	20%
	<i>Yes, from the users</i>	23%
	<i>No</i>	25%

Question (6)	Is usability an important element in your development projects?	
	<i>Yes, normally</i>	77%
	<i>Sometimes, depends on the project</i>	22%
	<i>No</i>	1%
Question (7)	If you would like to comment on question 6), please do so below:	
Question (8)	When will you include usability in requirements?	
	<i>Always</i>	72%
	<i>Only if usability problems emerge during the project</i>	10%
	<i>Only if the customer demands it</i>	15%
	<i>Only if we have an internal usability specialist available</i>	3%
Question (9)	How do you collect requirements for usability?	
	<i>Interviewing users</i>	68%
	<i>Best practice from earlier projects</i>	74%
	<i>Books, Internet resources</i>	20%
	<i>Other, please specify below</i>	12%
Question (10)	How important is usability requirements for the success of your projects? (Please indicate importance on a scale from 6 to 1, where 6 indicates very important and 1 quite unimportant)	
	<i>6 very important</i>	33%
	<i>5</i>	38%
	<i>4</i>	21%
	<i>3</i>	6%
	<i>2</i>	1%
	<i>1 Quite unimportant</i>	1%
Question (11)	How many users are typically engaged in usability testing?	
	<i>1–10 users</i>	67%
	<i>11–50 users</i>	21%
	<i>More than 50 users</i>	3%
	<i>We do not test usability</i>	9%
Question (12)	How do you select users for usability testing?	
	<i>Arbitrary sample of users</i>	5%
	<i>Representative sample of users</i>	40%
	<i>Own employees</i>	9%
	<i>Customer's employees</i>	23%
	<i>Other</i>	14%
	<i>Do not test usability</i>	9%
Question (13)	How important is usability testing for the success of your projects?	
	<i>6 very important</i>	14%
	<i>5</i>	23%
	<i>4</i>	31%
	<i>3</i>	19%
	<i>2</i>	6%
	<i>1 Quite unimportant</i>	5%
Question (14)	To which degree do you think that usability is integrated in your systems development method (whether you use a formal SDM or not)?	
	<i>6 To a large degree</i>	14%
	<i>5</i>	23%
	<i>4</i>	26%
	<i>3</i>	18%
	<i>2</i>	13%
	<i>1 Not at all</i>	3%

Question (15)	How many are working with systems development in your company?	
	1–5	46%
	6–10	10%
	11–50	35%
	More than 50	9%

References

- Beck, K., 2000. *Extreme Programming Explained: Embrace Change*. Addison-Wesley, Boston.
- Bekker, M.M., Vermeeren, A.P.O.S., 1996. An analysis of user interface design projects: information sources and constraints in design. *Interacting with Computers* 8 (1), 112–116.
- Beyer, H., Holzblatt, K., 1997. *Contextual Design: A Customer-Centered Approach to Systems Designs*. Morgan Kaufman Publishers, San Francisco.
- Boivie, I., Aborg, C., Persson, J., Lofberg, M., 2003. Why usability gets lost or usability in in-house software development. *Interacting with Computers* 15, 623–639.
- Boivie, I., Gulliksen, J., Goransson, B., 2006. The lonesome cowboy: a study of the usability designer role in systems development. *Interacting with Computers* 18, 601–634.
- Borgholm, T., Madsen, K.H., 1999. Cooperative usability practices. *Communications of the ACM* 42 (5), 91–97.
- Bygstad, B., Fagerström, A., Østensen, T., 2004. Exploring the relationship between software development processes and IT based business innovation. A quantitative study in Norway. In: *Proceedings of NOKOBIT 2004: Norsk konferanse for organisasjoners bruk av IT*, Stavanger, Norway.
- Catarci, T., Matarazzo, G., Raiss, G., 2002. Driving usability into the public administration: the Italian experience. *International Journal of Human-Computer Studies* 57, 121–138.
- Clegg, C., Axtell, C., Damodaran, L., Farbey, B., Hull, R., Lloyd-Jones, R., Nicholls, J., Sell, R., Tomlinson, C., 1997. *Information technology: a study of performance and the role of human and organizational factors*. *Ergonomics* 40 (9), 851–871.
- Constantine, L.L., 2002. Process agility and software usability: toward lightweight usage-centered design. *Information Age*, 1–10.
- Damodaran, L., 1996. User involvement in the systems designs process – a practical guide for users. *Behaviour & Information Technology* 15 (16), 363–377.
- Ehn, P., Löwgren, J., 1997. Design for quality-in-use: human-computer interaction meets systems development. In: Helander, M., Landauer, T.K., Prabhu, P.V. (Eds.), *Handbook of Human-Computer Interaction*. North-Holland.
- Fitzgerald, B., 1998. An empirical investigation into the adoption of systems development methodologies. *Information & Management* 34, 317–328.
- Gray, W.D., Salzman, M.C., 1998. Damaged merchandise? A review of experiments that compare usability evaluation methods. *Human-Computer Interaction* 13 (3), 203–261.
- Gould, J.D., Lewis, C., 1985. Designing for usability: key principles and what designers think. *Communications of the ACM* 28 (3), 300–311.
- Grudin, J., 1991. Interactive systems: bridging the gaps between developers and users. *IEEE Computer* 24 (4), 59–69.
- Grudin, J., 1993. Obstacles to participatory design in large product development organizations. In: Schuler, Douglas, Namioka, Aki (Eds.), *Participatory Design: Principles and Practices*. Lawrence Erlbaum Associates, New Jersey, pp. 99–122.
- Grudin, J., Poltrock, S., 1989. User interface design in large corporations: coordination and communication across disciplines. In: *Proceedings of CHI89*. ACM Press, pp. 197–203.
- Gulliksen, J., Boivie, I., Goransson, B., 2006. Usability professionals – current practices and future development. *Interacting with Computers* 18, 568–600.
- Gulliksen, J., Boivie, I., Persson, J., Hektor, A., Herulf, L., 2004. Making a difference – a survey of the usability profession in Sweden. In: Hyrskykari, A. (Ed.), *Proceedings of NordiCHI 2004*. ACM Press, pp. 207–215.
- Gulliksen, J., Goransson, B., Boivie, I., Blomkvist, S., Persson, J., Cajander, A., 2003. Key principles for user centred systems design. *Behaviour & Information Technology, Special Section on Designing IT for Healthy Work* 22 (6), 397–409.
- Gunther, R., Janis, J., Butler, S., 2001. The UCD Decision Matrix: how, when, and where to sell user-centered design into the development cycle. Available from: <http://www.ovostudios.com/upa2001/>, last accessed on October 18, 2006.
- Hutchings, A.F., Knox, S.T., 1995. Creating products customer demand. *Communications of the ACM* 38 (5), 72–80.
- Iivari, N., 2006. ‘Representing the user’ in software development – a cultural analysis of usability work in the product development context. *Interacting with Computers* 18, 635–664.
- ISO 9241-11, 1998. Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11 Guidance on Usability. International Organization for Standardization, Geneva.
- ISO/IS 13407, 1999. Human-centred Design Processes for Interactive Systems. International Organization for Standardization, Geneva.
- Jacobson, I., Booch, G., Rumbaugh, R., 1999. *The Unified Software Development Process*. Addison Wesley, Reading.
- Jerome, B., Kazman, R., 2005. Surveying the solitudes: an investigation into the relationships between human computer interaction and software engineering. In: Seffah, A., Gulliksen, J., Desmarais, M. (Eds.), *Practice in Human-Centered Software Engineering – Integrating Usability in the Development Process*. Kluwer.
- Karat, J., 1997. Evolving the scope of user-centered design. *Communications of the ACM* 40 (7), 33–38.
- Kujala, S., 2003. User involvement: a review of the benefits and challenges. *Behaviour & Information Technology* 22 (1), 1–16.
- Larman, C., Basili, V.R., 2003. Iterative and incremental development: a brief history. *IEEE Computer* 36 (6), 47–56.
- Larman, C., 2004. *Agile and Iterative Development: A Manager’s Guide*. Addison-Wesley.
- Microsoft 2004. *Microsoft Solutions Framework*.
- Mrazek, D., Rafeld, M., 1992. Integrating human factors on a large scale: ‘Product Usability Champions’. In: *Proceedings of CHI 92*. pp. 565–570.
- Nielsen, J., 1993. *Usability Engineering*. Academic Press, Boston, MA.
- Poltrock, S., Grudin, J., 1994. Organizational obstacles to interface design and development: two participant – observer studies. *ACM Transactions on Computer-Human Interaction* 1 (1), 52–80.
- Rational Corp, 2001. *RUP On-line*.
- Rauch, T., Wilson, T., 1995. UPA and CHI surveys on usability processes. *SIGCHI Bulletin* 27 (3), 23–25.
- Rosenbaum, S., Rohn, J.A., Humburg, J., 2000. A toolkit for strategic usability: results from workshops, panels, and surveys. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM Press, pp. 337–344.
- Sommerville, I., 2001. *Software Engineering*. Pearson Education, Harlow.
- Stapleton, J., 2003. *DSDM: Business Focused Development*. Addison-Wesley, Harlow.

- Vredenburg, K., Mao, J., Smith, P.W., Carey, T., 2002. A survey of user-centered design practice. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM Press, pp. 471–478.
- Truex, D., Baskerville, R., Travis, J., 2000. A methodical systems development: the deferred meaning of systems development methods. *Accounting, Management and Information Systems* 10, 53–79.