# Hello World! – Experiencing Usability Methods without Usability Expertise

Elina Eriksson, Åsa Cajander, and Jan Gulliksen

Uppsala University, Dept. of IT/HCI, PO Box 337, SE-75105 Uppsala, Sweden
{Elina.Eriksson,Asa.Cajander,Jan.Gulliksen}@it.uu.se

**Abstract.** How do you do usability work when no usability expertise is available? What happens in an organization when system developers, with no previous HCI knowledge, after a 3-day course, start applying usability methods, and particularly field studies? In order to answer these questions qualitative data were gathered through participatory observations, a feedback survey, field study documentation and interviews from 47 system developers from a public authority. Our results suggest that field studies enhance the developer's understanding of the user perspective, and provide a more holistic overview of the use situation, but that some developers were unable to interpret their observations and see solutions to the users' problems. The field study method was very much appreciated and has now become standard operating procedure within the organization. However, although field studies may be useful, it does not replace the need for usability professionals, as their knowledge is essential for more complex observations, analysis and for keeping the focus on usability.

**Keywords:** Field studies, ethnography, usability, user centered systems design, case study, public authority, sense making.

## 1 Introduction

Despite major research efforts within HCI and extensive practical adoption of HCI-methods, poor usability is still a major problem in applications resulting from in-house systems development, and causes frustration and stress in computer supported work [1]. There is a great need of usability activities, but fully incorporating usability aspects in the system development processes or in the organization is still difficult [2, 3]. HCI research stresses the importance of addressing usability early in the development process. However, design decisions that contribute to the usability of the end result happen throughout the development process. System developers[1] have great impact on the end-result through their needs to make design decisions throughout the system development process. Previous research on system developers in a public authority [4] examined how design decisions affecting usability were made, and on what grounds, showing that system developers regard their work as problem solving. The result of their work was judged by technical quality attributes rather than its

---

[1] In this paper system developer refers to a person mainly programming systems, although the work of the system developer may include modeling or design as well.

contribution to the end-user's work situation. The design of the user interface simply emerged without anybody consciously making the design decisions leading to the results. The results of these late design decisions often affect the usability of the system in a negative way, as they are not informed by the end-users' situation. However, system developers are not evil or ignorant in making these decisions. Rather, they do not know enough about the work of the end user. To get a better understanding of users and system usage, system developers need to apply usability methods. One such method is field studies.

## 2    Purpose and Justification

Our overall research objective is to make usability a key concern in practical systems development, and to impact the development processes to address usability issues. Field studies are widely accepted within Human Computer Interaction (HCI), and the general view is that the quality of the observations and documentation depend on the experience and sense making of the person conducting the field study. Experienced usability experts are thus likely to produce richer data and make observations of a different nature when conducting field studies, compared to novice users of the method. The practical reality in many organizations is that usability expertise may not be available, and if that is the case we need to find alternatives. Successful introduction of usability in organizations requires us to change attitudes and values of people working with systems development. One alternative might be to educate people with no previous usability expertise in usability methods hence impacting their sense making through experiences and understandings of the use situation. If usability methods are not regarded as useful by the developers, its prospect for successful deployment may be severely undermined [5]. Hence, this study examines how developers, with no previous knowledge of HCI, experience field studies in practice and discusses the possible implications it might have on system development. It does not focus on the documentation made in field studies, or the quality of observations, but on experiences and understanding gained by the system developers.

## 3    Theoretical Perspective

Examining the role and implication of usability methods in system development practice requires an understanding of human action and competence. Our research has been inspired by the perspective that human action is situated [6]. Consequently, our research must take place in practice. It consists of both "high, hard ground where practitioners can make effective use of research-based theory and technique" as well as "swampy lowland where situations are confusing 'messes' incapable of technical solution" [7]. Our research is based on a constructivist and interpretive perspective, where we create and understand our reality by using language through communication. Interpretations are flexible, situated, and socially constructed. We adhere to the quality criteria and principles established by Klein and Myers [8] as well as Rasmussen [9]. Such research based on case studies leads to contextual in-depth knowledge, which is hard to generalize. We as researchers, the context, the organization and the

conditions, under which the research takes place, color the results. However, even though the organizations and the findings are unique they are not uncommon and therefore we believe that the reader may find the knowledge gained applicable in other settings. Our perspective also partly originates from Participatory Design (PD) [10-12] which stresses the importance of involving users in the design process, arguing their right to be involved in development of their future use situation. PD has evolved during the last twenty years and has distanced itself from its political heritage towards a pragmatic view focusing on the quality of the user experience [11, 13]. Moreover, our perspective adheres to user-centered systems design (UCSD) [14, 15], which is also an international standard [16]. UCSD provides methods, roles, processes and techniques addressing usability and users' needs in systems development in practice. These approaches emphasize among other things the necessity of involving users, addressing usability, and understanding users' needs and work practices. For instance, they suggest studying users in real work situations in order to understand the activity-oriented view of work [17] reflecting what people do in their work to meet organizational and individual goals. These theoretical views guide us as researchers as well as informing the practice we try to improve.

## 4   Field Studies

A well-known method of gaining knowledge about the users' work situation is ethnographically inspired methods, e.g. field studies. Several practical variants of field studies have been developed, such as contextual inquiry [18], analysis of information utilization [19] and the ADA method [20]. Even though field studies are valued within HCI, they do not seem to be widely used in practice since many believe them to be too time consuming and producing too much data. Hughes [21] presents different forms of ethnographic methods that inform the design process, of which "quick and dirty ethnography" aims at doing shorter, more focused studies. It was based on insights that understanding complex organizations is not possible through a traditional ethnographic study given limited time frames and that fieldwork will provide no further useful insights that could aid the requirements engineering process. Another method, rapid ethnography [22] is based on the notion that that field methods are too time consuming. The process is made more efficient by limiting the research focus and scope, using key informants, capturing rich field data by using multiple observers and interactive observation techniques, and collaborative qualitative data analysis [22]. Previous research on field studies in practice has focused on adapting and streamlining field studies to the limited time scale of systems development. One example is Kujala's strategy labeled the "Trojan Horse" [23] where field study findings are presented in a familiar form, easy to implement in system development.

There is an ongoing discussion within HCI about the role of ethnography and ethnographically inspired methods. Most of the criticism is directed against the HCI research field rather than how the methods are used in practice. E.g. Dourish [24] argue that the HCI field more or less see ethnography as a method that generates requirements for systems design, thus overlooking the wider value of ethnography. Anderson [25] claims ethnography to be misunderstood, and that unfamiliar users of ethnography tend to think that it is easy to report back findings without interpretation,

thereby missing the analytical part of ethnography. This criticism can also be seen in the paper by Forsythe [26] where she presents 6 misconceptions about the use of ethnography in systems design. All in all these 6 misconceptions lead to the belief that ethnography is "common sense" that anyone can do, thereby miss the theoretical grounding and analytic work needed in order to produce reliable results. Ethnography leads to a rich in-depth material, and according to Plowman et al [27], it might be difficult for designers to make use of the results. All of the above-mentioned criticism is primarily focusing on the usage of ethnography or ethnography-inspired methods within research or by researchers to inform systems design. Bader and Nyce [28] aim their criticism mainly at practice and state that ethnography will not be a method that developers use frequently in systems development. Developers tend to "mistake themselves for their informants" thereby using data from ethnography to confirm their own beliefs. The authors' second reason is that developers' view of knowledge is rule bound and therefore the developers believe that work and social life can be explained with complex rules and principles. According to Nyce and Bader, developers will not value results stemming from ethnography or ethnographically inspired methods.

We argue that if field studies could be seen simply as a methodology to gather a scent of the use situation and the context of use, and not as a method to generate requirements or implications for design, we would be able to understand its value for developers in practice much better. We claim that it is crucial that developers broaden their perspective to make design decisions that may lead to better work environment and improved quality of work. In this sense, field studies can be a tool for developers to understand technology's impact on humans and work, and it might constitute a reflective tool that helps developers frame and reframe the problem space. One example of this would be to reconsider what system development is about, and perhaps see it more as designing future work, and less about creating perfect code.

## 5   The Case Setting

This research study was conducted in cooperation with the Swedish public authority that handles financial aid for students. The computer support used in the authority is mainly developed in-house, and a majority of these IT projects start because of changing legislations that influence the prevalent work practices. Consequently, time is a major factor controlling most IT development since the deadline of the implementation of a new legislation is fixed. IT projects have participants from different parts of the organization, as in a matrix organization, where each department has their role and responsibility. Officially, all projects are run according to methods provided by the business development department, and the IT architecture department. These include methods for acquisition, project management and systems development. In short, these methods provide a common framework, with descriptions of milestones, decision points, templates and role descriptions. Systems development is based on the waterfall model [29], and very few activities focus on usability. About two years prior to this study, a large 3-year action research project was launched within the organization with the purpose of increasing awareness and knowledge about usability and a healthy computerized work environment. The aim was to introduce and develop usability work practices based on our previous theoretical and empirical work [15, 30].

Focus is on increasing competence among all parties involved in developing computerized work through education and training, coaching and project cooperation.

## 6  Method

Our research is based on an action research perspective, see for example Rasmussen [9] and Avison et al [31]. As an action research project, it has two goals as described by McKay and Marshall [32], i.e. one research interest and one problem solving interest for the organization in which the research is performed. The interest in system developers and field studies is one good example where the organization and their concerns have guided the research, as they wanted to find ways to use usability methods despite the fact that there were few usability experts in the organization. We participated in project activities as members of the project group, at the same time as we observed certain aspects of the interaction and communication as researchers. In this study, data was generated during four months through participatory observations, a qualitative survey, interviews and the developers' own field study reports. The events and data generation methods are described in more detail below and summarized in Table 1.

**Table 1.** Data generation methods and events

| Usability training course | Pilot project |
|---|---|
| • Participatory observation during course; field notes<br>• Feedback survey after field study<br>• Field study reports<br>• Course evaluation<br>• Interview with Brian, system developer | • Participatory observation in project meetings; field notes<br>• Participatory observation debriefing meeting after field study; field notes<br>• Interview with Scott, system developer |
| • Interview with John, system developer ||

During fall 2006, one of us planned and implemented a usability training course for developers. The course was three days long with 47 participants. The first day contained basic knowledge on general usability, quality attributes and a practical introduction to field studies, based primarily on the ADA method [20]. The second day the developers planned conducted and documented field studies in pairs. A field study feedback survey was distributed to the 47 participants in the usability training course, of whom 36 responded. The aim of the survey was to get information about how developers experienced the field study. The feedback survey consisted of 10 questions ranging from general questions about background and previous experience to specific questions about the advantages and disadvantages experienced with field studies. The

feedback survey contained multiple choices and open-ended questions in free text form. The third day the results obtained in the feedback survey was discussed and basic introduction to design workshops and evaluation methods were taught. Furthermore, we got access to the field study documentation that the developers had written, as well the results from the course evaluation made by the authority. In addition to the researcher giving the course a second researcher participated, doing participatory observations, which resulted in field notes.

The field studies the developers in the usability training course performed were not part of any ongoing project, although some of the developers chose to look at systems they had been or were developing. To get another perspective of how developers experience field studies, we included results from a software development project into this study, which was a pilot for testing usability methods. In this paper this project is referred to as the pilot project. The project was followed from pre-study to project start between August and December 2006. None of the participants in the project had any previous HCI knowledge or experience, and therefore they participated in the usability training course described above. After the course they conducted field studies in the pilot project, and discussed the results at a debriefing meeting. One of us followed the pilot project and made participatory observations. This resulted in field notes. Moreover, we interviewed three developers in order to further understand the their experience of field studies. The developers were selected from different perspectives, one had done field studies in both the usability training course and the pilot project (John), one had only done a field study in the pilot project and had not participated in the usability training course since he was a consultant and not employed by the public authority (Scott) and one had only done a field study in the usability training course (Brian). The 45-minute interviews were semi-structured, used an interview guide and were audio recorded. The feedback survey was analyzed and summarized into written form with emergent themes highlighted. The summarized feedback survey was presented at the last course day, and the different themes were discussed with the course participants. These discussions were partly recorded, and partly written down in participatory observations. The different participatory observations all ended up in hand written field notes, and these field notes were read thoroughly and used as both inspirations for questions in the interviews as well as material for the article. The course evaluation and the field study reports written by the developers were read through and interesting parts were highlighted and categorized into themes. The interviews were listened through and subsequently transcribed, printed and cut in pieces, color marked and sorted into themes. The themes in the feedback survey, field study reports, course evaluation and interviews were compared and discussed between all three authors. When writing up the study all names were altered, and quotations used are not transcribed verbatim.

## 7   Results

The results mainly focus on how developers experience field studies. When defining their formal role in free text, the majority of the respondents (26 out of 36 that answered the survey) wrote system developer. However, 4 were system specialists, 3 technology development leaders, 1 web designer, 1 consultant and 1 process

developer. The interviewees as well as the participants that conducted field studies in the pilot project were system developers.

## 7.1 Holistic Overview versus Concrete Problems

The interviewees were asked if anything from the field studies could be used in their work. The interviewees mention the importance of gaining a holistic overview rather than pointing at concrete problems. The holistic view can be divided in two aspects. First, the overview of the system development project, that is, knowing the common goal of the project, what is happening and take part in all the stages of the project. Scott gives an example: *"The best thing is when you've been there so you know what it is all about /…/ so that you know what we are supposed to do, where we're heading."* Second the overview gives knowledge about core business. The developers have a wish to understand the context and work of case handlers[2] in order to create good systems. For example, when asked about what he needs in order to do a good job, Brian first states that he needs good programming tools and then core business knowledge. When prompted for clarification of what he means by core business knowledge he says: *"I need to understand where I am, what I'm doing and why. Yeah that's it, to understand the whole flow, to understand the process when people work"*

A majority of the answers in the feedback survey concern different aspects of a holistic view. They report insight into the system functionality and its interfaces with other systems, insights into what the user's work actually is about. Other insights relate to the users' work situation, e.g. that users felt trapped and couldn't receive essential help from anywhere and that the users don't see the possibility of influencing the system design. Moreover, some respondents where surprised at how stressful the case handlers described their work with the computer systems, and the extent to which work was manual, dealing with a lot of printed material. One respondent was surprised at the extent to which users used informal information, e.g. "reading between the lines" and the degrees of freedom in how people made use of the computer system. Interestingly, the respondents also noted several things concerning the user's work process, for example difficulties built into the work due to a mismatch between business processes and the needs of the work tasks. The work process was also discussed during the debriefing meeting after the field studies in the pilot project. Participants had seen problems that were hard to classify as work process problems or non-supportive IT-product.

In the pilot project participants found more concrete problems than problems related to holistic view. At the debriefing meeting, large amounts of problems were discussed ranging from a need of a weekly calendar for case handlers (someone had actually glued a calendar onto the screen of the computer) to the intense use of the mouse (since the instructions for shortcut keys where hard to find). During the interview, John started to recount different details that he would like to change, certain aspects of the graphical user interfaces that forced the user to do a lot of work with the mouse etc, which is also something he had written in his field study report. When prompted if he could change these things in the current project, he said no, it was not planned, and there was no time for this. All in all, the participants in the pilot project

---

[2] Case handler refers to a civil servant working at a public authority with case handling.

were overwhelmed with all the data they had collected, and wondered who should receive a description of the needs for identified improvements. Incorporating the problems in the pilot project was impossible due to time constrictions.

The feedback survey shows that some developers gained knowledge about the more detailed uses of the system. As opposed to the pilot project participants, they were surprised at the extent to which shortcut keys were used, although it should be noted that they were all observing use of different systems. Another finding was the practical difficulties making use of scanned information and documents and the effect this had on the case handlers' work. Three respondents also noted the extent to which text templates needed to be reworked for every situation.

## 7.2   Seeing without Gaining an Insight

Some developers report explicitly that they did not see anything of value during the field studies. However, when looking through the material, several things could be discerned that trained usability practitioners would notice and bring up. From a sense making point of view this reveals a discrepancy between what developers see, and what they find relevant. There is more to it than simply see and record as they must also gain insight. Brian, for example, was negative to involving end users in systems development and doing field studies. He expressed that it was interesting doing the field studies to see if the case handler worked the way the development team had thought they should work. But he concluded that the field studies didn't give him anything that he could use in his work as a developer.

> *"Researcher: But did the field study give you something*
> *you could use when you are developing?*
> *Brian: No*
> *R: And you saw nothing you could improve or something like that?*
> *B: No*
> *R: Can you imagine another situation where you could have*
> *gained something from a field study?*
> *B: No"*

In the light of this, Brian's co-authored field study report is interesting. It states two concrete problems besides those a trained usability practitioner could find. First, the case handler did not have time to finish the case she was working with, whilst having the customer on the phone. She recorded data on post-its to finish later, which she found easier than registering everything in the system. However, even writing information on post-its seemed difficult to manage during the call. Second, there was not enough information in one of the windows in use, and the developers gave a brief solution to the problem. Later in the same interview, Brian explained that he didn't think his field study was beneficial because the system, (that he had been developing) had been in use for a long time. He would have preferred to look at the use of the system after 3-4 months. Furthermore few reports contain clear statements on what is wrong in the computer supported work. It is as if they have not really reflected on what is wrong, but on the other hand it was not made explicit in the usability training course that they should come up with any solutions in the field study report. Moreover, when problems are found, only two of the reports suggest a solution and in some cases the only problems that are mentioned are the ones that the case handler has pointed out.

### 7.3 Responsibility

Developers were prompted to discuss what to do if having to make a design decision that might affect the user when programming. Two of the interviewees, Scott and Brian, said they talk to the other project group members. Usually the question ended up at the project management level. John said he had not thought about whether his decision would affect the end-user in the end. But, when thinking about it, he would like to talk to the user in that situation, since they were the ones that would use the system in the end: *"We have done that before. 'You can decide since you're the one who will work with the system in the future.' Then you put the responsibility on them somehow"* Brian was less eager to give responsibility to the users. In his opinion, it was useless to talk to real end-users, since they are not able to express what they want and they do not know the possibilities available. *"We are as good as they are at guessing! And I don't buy all these things about working with the users because they know how things work. Because I have been talking to users and I know the rules better than they do since they are used to doing things in a certain way. I who don't work with these things say that I have read the rules and these are the rules. Then I can do it easier somehow. I wouldn't say that I am better than they are at saying what they want. But still somehow we guess at what they need and they are not better than we are at expressing that need. Of course there are some things that they can tell us, but I don't think it is a good idea."*

### 7.4 Documentation and Templates

Developers generally did not want an interview template as it might disturb the interview preparation, and risk including too general questions, hence being less usable for the developers. However, a quick interview guide might come at hand. When discussing the documentation, the developers favored hand-written notes instead of a document template. However, in their opinion, a template facilitates incorporation into the development process and makes the documents easier to read. The interviews also show that written program specifications are less important when having a holistic view. Scott: *"If you know where we're heading, broadly so to speak, yes that is good starting point. That's what's really important. And the specification, what's written, is less important then."*

### 7.5 Feedback and Motivation

In the field study survey feedback, developers write about feedback and contact with real end-users. Some could see future benefits from doing field studies, such as: *"Good to establish some real contact with end-users that we could make use of in the future"*. Furthermore, one of the respondents found it especially motivating to *"see things that I can actually change in the system construction to improve the user's work situation"*. In the interviews the developers experienced that the field studies had given them feedback and motivation in their work. They wanted more feedback from users in their work, especially critical feedback. John mentioned that there is usually no user participation early on in the development process, and the case handlers involved in the test phase in the end of the process, do as they are told, and do not give enough feedback. And then he concluded that it was too late anyway to get

feedback from the case handlers doing tests, since there were no chances of modifying the system after the test phase. He had done so once and been criticized by the development project. In the interview Brian expressed dissatisfaction with the feedback from the users. He got too little feedback and wanted more. According to him, users do not complain enough, or the criticism does not reach developers. The developers involved in the pilot project were also truly motivated by the field studies. During the debriefing meeting all the participants talked animatedly about their field studies and seemed to have enjoyed it, one exclaimed when it was his turn to talk that: *"Directly after I thought: Damn, I should visit more case handlers!"*

## 8  Discussion

### 8.1  Holistic Overview

An appropriate understanding of the context of use is one of the foundations of a user-centered development process [16]. Our findings show that one of the major benefits of having developers do field studies is the increased knowledge and awareness of the context of use. This is also consistent with surveys of user-centered design in practice [33, 34]. The context of use is one part of a holistic overview, which was emphasized in all the data from the study. But the fact that developers talked about holistic overview, without noticing concrete problems to the same high degree, might have other explanations. Developers work in projects where time is short and where they have no opportunity to deal with the encountered problems. To make sense of the method they might react to these cues; lack of time and no possibility to change things, and when asked about the use of the method, the most plausible answer they have is that the benefit of the method is the holistic overview [35]. Other aspects that might affect what the developers gained from the field studies are their lack of experience and almost no HCI knowledge. The field studies were conducted during a day, or part of a day, and in some cases it was a system that they had not built themselves. Facing a complex work situation without experience, developers are overwhelmed with information, which lead them to notice the holistic overview before details. One could argue that we influenced the developers in their view of the benefits of the method. However, the course was filled with tips and discussions about the details, as well as the holistic overview of field studies to address this bias.

### 8.2  Developers' Background and Knowledge

When developers, often trained in a positivistic tradition, learn about field studies as a usability method they often position themselves as neutral observers, expressing a concern that they will inevitably distort the data and introduce subjectivity as their presence will affect the work situation. This is consistent with others observations, as *"many people confuse observations with inference"* [36]. As in this example from the survey: *"We need better tools for doing this, in order for us not to lead the users in a specific direction."* Qualitative methods, such as field studies, are based on a different perspective, where the subjective and interpretive elements are foundations of human

understanding. The developers' understanding of the setting and context depends on his/her perspective of the world which is, shaped by background, social position, gender, etc. In relation to the results, we would like to elaborate around three different kinds of knowledge that affect what the developers see or not see during the field studies, HCI or ethnographical knowledge, knowledge about the core processes in the organization and knowledge about the IT-systems. It is quite clear that the developers lack HCI or ethnographical knowledge. One easily perceived example of how differently a developer interprets the situation from a usability expert is Brian, who believes that users become "ruined" and constitute bad respondents when they have worked with a computer system for some years. These results are consistent with that of [36] as the observer "fail to look for information that may contradict or challenge assumptions". What is also apparent in the results is that the developers did not see beyond the user and the context as place and situation. None of the developers elaborated around culture etc. of which those with ethnographical or anthropological background would argue is the central outcome from a method like field studies [25, 26]. Nor did they focus on "implications for design" [24], that is concrete things to be considered during the development. Instead they ended up praising the holistic overview that they gained from the studies. Knowledge about the core business is apparently something that the developers seek, and the field studies are one way of gaining this knowledge. However, some of the developers have core business knowledge and has worked a long time in the organization. Other developers noticed that the case handlers did not work in the way they had predicted, however, they could not say if it depended on the work process or a faulty IT-system. Finally, knowledge about the IT-system, and the underlying technology, makes the developers see small details during the field studies that can easily be changed.

### 8.3   Field Studies Helping Knowledge Transfer

Several of the developers expressed that they would bring the experience from the field study into development projects in the future. However, in development projects where time and resources are scarce, not all systems developers will get the chance to visit case handlers in every project. Furthermore, usability experts will still make field studies and report back their findings to the development group. As in many knowledge intensive organizations, there is a need to codify knowledge, and transfer the knowledge to those who has not been present when the knowledge was created [37]. Conducting field studies, albeit not in all projects, will give the developers a framework, which will help them to understand the information and knowledge conveyed to them by others. Or as Davenport and Prusak states it [37]: *"Knowledge is more likely to be absorbed if it adheres to the listeners' sense of ground truth, … and is placed in a context of frame that is at least partly shared by its audience"* .

### 8.4   Motivation

Field studies have the potential to motivate developers, and reduce their motivation. It motivates developers in that they understand that what they do affect others, and that they see the practical consequences of their work. We were intrigued by the joy the

developers expressed over the field studies, and the enthusiasm most of them showed while talking about the field studies. Several expressed wishes to go back and do something about the problems that they encountered, because they know quick ways of dealing with the problems. But unfortunately the current project structure does not allow them to deal with problems they encounter, as they only may address problems that they have allocated time for. If they want to make changes based on their observations they must propose this as new system development projects that need to go through the annual work planning and prioritization process. This is frustrating and therefore the field studies might reduce their job motivation. The interview data indicates a wish for overview and early participation as developers wish to have a stimulating work. Simply being "coders" is not as stimulating as being a part of the whole process. Participation in different stages of the process gives some variation to their work. Moreover Brian states that: *"The requirement specification should be more on an overview level and not as detailed and controlling, because if that would be the case, whoever had done the requirement specification could do the coding as well, translating from a sheet of paper to a program, which is something anybody could learn. No, I'd much rather see a requirement specification on a more overview level, and if you do understand the business processes, you are able to turn it into a great system"*. Clearly Brian wants to be more than just a "coder"; rather he would like to take part in the design of the system. Field studies done in the pre-study or early in the system development process would give him some of the information needed to understand the business processes and hereby design a usable system.

## 8.5   Practical Consequences and Possible Implications

**Higher Focus.** We believe that field studies can give usability a higher focus in the organization. However, this implies that the methods with deliverables need to be specified in the systems development process; otherwise the projects will not allocate time and resources to it. Field studies would help promote the user situation as a part of the project goals and could then potentially lead to increased focus on quality. Unfortunately we have seen that usability has come to be synonymous with field studies in the organization. There is a risk that the sustainability of these methods in the long run will be affected if the usability issues are not safeguarded throughout the project. Yet another practical consequence is that the developers through field studies meet with case handlers and establish a contact that they can use throughout the development. This will make it possible for developers to contact case handlers for small questions that even though small, can impact the end result.

**Potential Risk.** Can the fact that developers are trained to do field studies, design and evaluation, but have little HCI knowledge potentially be harmful? There are always potential risks with adopting new approaches like this, as the organization might conclude that since these usability methods are potentially useful for everybody, then we do not need any usability expertise. These findings are similar to [38], who concludes that there seems to be an ideal that the HCI practitioners are not needed at all since every member of the system development project is an HCI practitioner. However, as [38] discusses "If HCI work is everybody's then nobody's responsible".

Does increased knowledge about users lead to increased power for developers in the development process, as knowledge is power? Gillian [39] indicate that user involvement can be used as a political tool within organizations and a tool for developers to "manage their decision making process". In our study, developers showed few tendencies of using field studies to promote their own ideas. On the other hand, future misuse of this power is still possible. Some developers were surprised at the multiple ways in which technology was used. Potentially this surprise, and the discussion afterwards, make them question their own technological deterministic perspective, hence realizing that users are not passive recipients of technology but actors [40] who create the circumstances, context and consequences of technology in use. Practice gives meaning to technology, and a good work environment is created in the situated experience of technology in context. We argue that a small insight into the world is better than none. However, Forsythe [26] criticizes this view: "…surely some knowledge of a situation is better than none. The problem is that in ethnography as in some other pursuits, a little knowledge can be a dangerous thing: superficial social research may confer the illusion of increased understanding when in fact no such understanding has been achieved". We argue that developers are not involved in research, and furthermore they are not trying to do ethnography.

**Usability Expert.** We do not discard the need for a usability expert, who should have the knowledge and experience to challenge assumptions that developers might have. The usability expert's knowledge is essential for making more complex observations and for keeping the focus on users and usability throughout the development project. We agree with Anderson, that: "This is not to say that getting to know users and their knowledge and practices is unnecessary or irrelevant or that observational fieldwork and impressionistic reportage can be of no value in this. Far from it! It is simply that you do not need ethnography to do that; just minimal competency in interactive skills, a willingness to spend time, and a fair amount of patience." [25]

**Impact on practice.** Most of the developers who conducted field studies as a part of the course experienced it as useful, interesting and motivating. Our mission as action researchers is to make usability have a greater impact in organizations. One way of doing so is to involve usability expertise. But from previous studies we know that usability professionals may not have the impact in practice that one would have expected [2]. Or as Siegel and Dray put it: "UCD professionals who focus on doing "studies" as opposed to generating designs and products, will always be perceived as peripheral." [41]. We believe that the HCI research community partly is to blame for this. It may have been too focused on quantitative empirical studies, and scientifically validating the findings rather than making an impact on practice. Hence, HCI has produced a vast number of analysis techniques and usability evaluation methods, mainly to be applied by usability professionals. From a scientific point of view this is excellent, but does it really contribute to the development of practice? As HCI researchers we should be more concerned about making our methods, tools and knowledge used by practitioners who generate designs and products, and some of these practitioners are developers. As an interesting epilogue, we can report that inspired by the field studies conducted by the developers, the case handlers wanted to

see how the developers worked. In the work exchange program organized by the union, case handlers visit headquarters and the developers' work setting, in order to get an understanding of their situated work and not only other case handling offices.

# References

1. Ceaparu, I., Lazar, J., Bessiere, K., Robinson, J., Shneiderman, B.: Determining Causes and Severity of End-User Frustration. International Journal of Human-Computer Interaction 17, 333–356 (2004)
2. Boivie, I.: A Fine Balance: Addressing Usability and Users Needs in the Development of IT Systems for the Workplace. Acta Universitatis Upsaliensis, Uppsala 85 (2005)
3. Rajanen, M., Iivari, N.: Usability Cost-Benefit Analysis: How Usability Became a Curse Word? In: Baranauskas, C., Palanque, P., Abascal, J., Barbosa, S.D.J. (eds.) INTERACT 2007. LNCS, vol. 4663, pp. 511–524. Springer, Heidelberg (2007)
4. Boivie, I., Gulliksen, J., Göransson, B.: It's all in a Days Work of a Software Engineer HCI International (2003)
5. Riemenschneider, C.K., Hardgrave, B.C., Davis, F.D.: Explaining software developer acceptance of methodologies: a comparison of five theoretical models. IEEE Transactions on Software Engineering 28, 1135–1145 (2002)
6. Suchman, L.: Plans and Situated Actions: The Problem of Human-Machine Communication. Cambridge University Press, Cambridge (1987)
7. Schön, D.: The Reflective Practitioner - How Professionals Think in Action. Ashgate Publishing, Aldershot (1983)
8. Klein, H., Myers, M.: A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems. MIS Quarterly 23, 67–94 (1999)
9. Rasmussen, L.B.: Action research—Scandinavian experiences. AI & Society 18, 21–43 (2004)
10. Greenbaum, J., Kyng, M.: Design at work: cooperative design of computer systems. Lawrence Erlbaum Associates, Inc, Mahwah (1992)
11. Asaro, P.M.: Transforming society by transforming technology: the science and politics of participatory design. Accounting, Management and Information Technologies 10, 257–290 (2000)
12. Bødker, S., Iversen, O.S.: Staging a professional participatory design practice: moving PD beyond the initial fascination of user involvement. In: Second Nordic conference on Human-computer interaction, pp. 11–18. ACM Press, New York (2002)
13. Bødker, S., Ehn, P., Sjögren, D., Sundblad, Y.: Co-operative Design - perspectives on 20 years with The Scandinavian IT Design Model. In: NordiCHI (2000)
14. Norman, D.A., Draper, S.W.: User Centered System Design; New Perspectives on Human-Computer Interaction. Lawrence Erlbaum Associates, Inc., Mahwah (1986)

15. Gulliksen, J., Goransson, B., Boivie, I., Blomkvist, S., Persson, J., Cajander, A.: Key principles for user-centred systems design. Behaviour and Information Technology - BIT 22, 397–409 (2003)
16. ISO 13407: Human-centred design processes for interactive systems. International Organisation for Standardization, Geneva (1999)
17. Sachs, P.: Transforming work: collaboration, learning, and design. Communications of the ACM 38, 36–44 (1995)
18. Beyer, H., Holtzblatt, K.: Contextual Design: Defining Customer-centered Systems. Morgan Kaufmann Publishers Inc., San Francisco (1998)
19. Gulliksen, J., Lif, M., Lind, M., Nygren, E., Sandblad, B.: Analysis of Information Utilization(AIU). International Journal of Human-Computer Interaction 9, 255–282 (1997)
20. Åborg, C., Sandblad, B., Gulliksen, J., Lif, M.: Integrating work environment considerations into usability evaluation methods—the ADA approach. Interacting with Computers 15, 453–471 (2003)
21. Hughes, J., King, V., Rodden, T., Andersen, H.: Moving out from the control room: ethnography in system design. In: 1994 ACM conference on Computer supported cooperative work, pp. 429–439 (1994)
22. Millen, D.R.: Rapid ethnography: time deepening strategies for HCI field research. In: Conference on Designing interactive systems: processes, practices, methods, and techniques, pp. 280–286. ACM Press, New York (2000)
23. Kujala, S.: User involvement: a review of the benefits and challenges. Behaviour & Information Technology 22, 1–16 (2003)
24. Dourish, P.: Implications for design. In: Proceedings of the SIGCHI conference on Human Factors in computing systems, pp. 541–550. ACM Press, New York (2006)
25. Anderson, R.J.: Representations and requirements: the value of ethnography in system design. Human-Computer Interaction 9, 151–182 (1994)
26. Forsythe, D.E.: "It's Just a Matter of Common Sense": Ethnography as Invisible Work. In: Computer Supported Cooperative Work (CSCW), vol. 8, pp. 127–145 (1999)
27. Plowman, L., Rogers, Y., Ramage, M.: What are workplace studies for?: The fourth conference on European Conference on Computer-Supported Cooperative Work, pp. 309–324 (1995)
28. Bader, G., Nyce, J.M.: When only the self is real: theory and practice in the development community. Journal of Computer Documentation 22, 5–10 (1998)
29. Royce, W.W.: Managing the development of large software systems: concepts and techniques. In: 9th international conference on Software Engineering, pp. 328–338 (1987)
30. Sandblad, B., Gulliksen, J., Aborg, C., Boivie, I., Persson, J., Goransson, B., Kavathatzopoulos, I., Blomkvist, S., Cajander, A.: Work environment and computer systems development. Behaviour and Information Technology - BIT 22, 375–387 (2003)
31. Avison, D., Lau, F., Myers, M., Nielsen, P.A.: Action Research. Communications of the ACM 42, 94–97 (1999)
32. McKay, J., Marshall, P.: The dual imperatives of action research. Information Technology and People 14, 46–59 (2001)
33. Vredenburg, K., Mao, J.Y., Smith, P.W., Carey, T.: A survey of user-centered design practice. In: SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves, pp. 471–478. ACM Press, New York (2002)
34. Gulliksen, J., Boivie, I., Persson, J., Hektor, A., Herulf, L.: Making a difference: a survey of the usability profession in Sweden. In: Third Nordic conference on Human-computer interaction, pp. 207–215. ACM Press, New York (2004)
35. Weick, K.E.: Sensemaking in Organizations. Sage, Thousand Oaks (1995)

36. Wixon, D.R., Ramey, J., Holtzblatt, K., Beyer, H., Hackos, J.A., Rosenbaum, S., Page, C., Laakso, S.A., Laakso, K.P.: Usability in practice: field methods evolution and revolution. In: SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves, pp. 880–884. ACM Press, New York (2002)
37. Davenport, T.H., Prusak, L.: Working knowledge. Harvard Business School Press, Boston (1998)
38. Iivari, N.: Understanding the work of an HCI practitioner. In: 4th Nordic conference on Human-computer interaction: changing roles, pp. 185–194. ACM Press, New York (2006)
39. Gillian, S.: The Work of IT System Developers in Context: Organizational Case Study. Human-Computer Interaction 13, 37–71 (1998)
40. Bannon, L.: From Human Factors to Human Actors: The Role of Psychology and Human-Computer Interaction Studies in System Design. In: Greenbaum, J., Kyng, M. (eds.) Design at Work: Cooperative Design of Computer Systems, pp. 25–44. Lawrence Erlbaum Associates, Hillsdate (1991)
41. Siegel, D., Dray, S.: Living on the edges: user-centered design and the dynamics of specialization in organizations. Interactions 10, 19–27 (2003)