Usability of Mobile Applications
Dissemination of Usability Engineering in Small and Medium Enterprises

Britta Fuchs, Thomas Ritz and Jakob Strauch
Faculty of Electrical Engeneering and Information Technology, FH Aachen, Eupener Str. 70, 52070 Aachen, Germany

Keywords: Usability Engineering, Mobile Applications, SME, User-centred Development.

Abstract: This paper starts from the idea that mobile enterprise software has great potential in the future but needs to fulfil usability requirements to be successful. Following mobile enterprise software Mobile enterprise software is explained with the established software engineering processes. Related to this topic usability engineering is presented with relevance to the utilization in small and medium-sized enterprises (SMEs). The relevance of the target group SMEs is demonstrated using the example of Germany. The integrated user-centered method for mobile enterprise software development integrating usability is presented. The paper closes with an analysis of the needed future research in this field.

1 INTRODUCTION

Mobile enterprise software systems increasingly turn out to be suitable for integrating external workers of small and medium-sized enterprises (SMEs) into operational information logistics. However, in terms of usability mobile software systems differ have features that were responsible for the failure of many such projects in the past.

This paper will first explain mobile enterprise software and usability engineering. Based on this the usability criteria are highlighted that need to be considered when developing mobile enterprise software for SMEs. Then, the requirements for an adequate approach to take usability criteria and needed tools into account are derived.

2 MOBILE ENTERPRISE SOFTWARE

The concept of enterprise software is established as a diffuse summary of business application systems. The goal of mobile enterprise software is the involvement of employees who do not work at a fixed location. Especially in machine and plant construction, the provision of services is a task that is often done by the field staff.

The following definition of mobile enterprise software (Ritz, 2003) enlarges this idea with important requirements:

"Mobile enterprise software is the application of business software solutions in mobile use

1. on appropriate mobile devices
2. with appropriate functionality
3. based on appropriate data currentness. "

From this definition immediately result a set of requirements concerning a development approach of mobile enterprise software solutions (see Table 1). Initial approaches to consider localities in the requirements analysis and related infrastructure issues are integrated e.g., in Mobile Process Landscaping (Köhler and Gruhn, 2004). There are also many variants and extensions of UML (e.g. M-UML Saleh and El-Morr, 2003, agile M-UML Baumeister, 2005) involving mobile aspects, but address the modeling of requirements less than modeling contexts for autonomous mobile systems and user profiles.

Our research group was able to develop a tool that extended UML with the stereotype location and that makes characteristics of the user (actor), e.g. about his mental abilities ascertainable (Ellerweg and Ritz, 2008a, Ellerweg and Ritz, 2008b). Especially for the development of mobile service solutions there is no adequate modeling support to systematically record and document any special requirements during the requirements analysis.

Fuchs B., Ritz T. and Strauch J..
Usability of Mobile Applications - Dissemination of Usability Engineering in Small and Medium Enterprises.
DOI: 10.5220/0004068802720277
In *Proceedings of the International Conference on Data Communication Networking, e-Business and Optical Communication Systems* (ICE-B-2012), pages 272-277
ISBN: 978-989-8565-23-5

Table 1: Core elements of mobile software engineering.

1.	Location and context must be considered in the specification of software. In particular, a change of location where an application is used causes alterations e.g. on the available network infrastructure or the environment situation (noise, light ...). Such requirements must be imposed within the software engineering process and their fulfillment need to be checked in test scenarios
2.	Interaction and basic paradigms differ from stationary to mobile applications. Working with mobile IT systems is often characterized by short and frequently interrupted activities (Salmre, 2005). The applications must be adjusted accordingly. It should be regarded that in mobile business processes often people with a limited affinity for computer applications are involved. The specific requirements have to be collected as part of the software engineering process.
3.	Multitude of possible devices are available with very different characteristics (Hansmann, Merk, Nicklous, & Stober). Selection and use of these characteristics has to be central element of the software engineering process.
4.	A variety of architectural options is available specifying in particular whether data is stored on the mobile device or stored centrally (Roth, 2005). The definition of a suitable architecture is therefore a central element of a software engineering process. Doing so it must be borne in mind that architecture components such as networks are not consistently available or are encumbered with additional costs.

The software engineering process has to methodically identify main activities the user has to fulfill in parallel (e.g. repair work) that may interrupt the use of the mobile system (Salmre, 2005). To be able to cover the requirements of the individual user as good as possible, the particular user abilities have to be integrated into the development process. Additionally the software system should be created in close consultation with the users.

To meet the demand for customized functionality, user-centered design methods have proved themselves (Norman & Draper, 1986). They as well lived up to the expectations when it comes to integrating customers beyond interaction into software projects as an important input factor.

The customer as most important input factor is consulted in *iterations* to (re-)define both, the requirements and the acceptance of sub-projects. The most important representatives of iterative software development methods are Extreme Programming (Beck & Andres, 2004), Crystal Methodologies (Cockburn, 2005) or Scrum (Schwaber, 2004). Usually the methods focus the engineering-based approach to software development and management of resources (see e.g. Cooper, Cronin, & Reimann, 2007).

These approaches aim to handle user-centered functional requirements. In contrast measures of usability engineering (Mayhew, 1999) target to integrate the definition of usability criteria (effectiveness, efficiency and satisfaction) (DIN EN ISO 9241-1) into the development process and as well to verify those criteria. Our research aimed at developing an integrated method combining both approaches, extreme programming and usability engineering.

3 USABILITY ENGINEERING

Usability is understood as the degree of effectiveness, efficiency and satisfaction in which users can perform certain tasks with an application in a given environment (e.g., office environment) (see DIN EN ISO 9241-1).

In this context process models have been established that systematically consider usability as a performance measure, covering the entire product life cycle. These process models can be summarized under the term "usability engineering" (see e.g., Mayhew, 1999).

While user interface design addresses mainly visual user interface design, usability engineering, describes both, the development of interactive systems and as well the design of appropriate work practices and interactions of users in the specific context of use. Therefore, usability engineering influences the design surface, but is not limited to it.

More advanced software development is increasingly beyond the aspect of usability and is dedicated to the field of *user experience*. User experience describes the "perceptions and reactions of a person resulting from the actual and / or the expected use" of an application (DIN EN ISO 9241-210).

User experience adds a clearly hedonic component. It is characterized by aesthetics, fun, functionality, enjoyment and personal development (Hassenzahl, 2004) but as well by the brand image. The latest developments such as the success of the Apple iPhone illustrates that even supposedly non-interactive design domains benefit from user experience (Klauser and Walker, 2007).

On the one hand, user experience affects interaction and interface design. On the other, user documentation and support service as well as maintenance have to be considered. In order to meet these requirements the users strengths, limitations, preferences and expectations have to be taken into account (DIN EN ISO 9241-210).

With regards to medium-sized companies employing software the following key findings were identified by the research project "Usability in Germany" (Woywode, Mädche, Wallach, & Plach):

- In certain software employing enterprises usability is an explicit purchase criterion previous to other criteria such as price or functionality.
- Software with a high degree of usability is created primarily by software producers with a strong user orientation. These criteria can be found in companies where management has a positive attitude towards usability and where extensive knowledge regarding usability is available within the company.
- In addition to the perceived customer requirements the interaction in the organizational area contributes significantly to mental attitude and knowledge development.
- Software producers who assess a high usability degree of their products recorded an above-average corporate success in recent years (as measured by customer satisfaction and sales performance).

4 TARGET GROUP SMALL AND MEDIUM ENTERPRISES

In the online survey "Questionnaire User Need" (FUN), SMEs were asked about their need for usability. The survey concludes that in medium-sized businesses a general acceptance of usability services exists. Once companies were aware of usability issues, they were more open to external services.

A majority of the SMEs surveyed considers that the application of usability methods in earlier stages of development of suitable and desirable. More than 70 % of the respondents subjectively assessed their own products as sufficient easy to handle. That's why they could not find a reason to increase the cost of usability measures in the future (Brau et al., 2011).

While 1,000 companies were asked to participate in the survey, the results are based on the records of only 41 companies. Since it is possible that the other companies did not respond because they are uncertain, how to deal with usability the findings of the survey apply only as indicators. A large proportion of medium-sized German software companies is still in its infancy with respect to this issue (Woywode et al.).

Since a couple of years, the relevance of usability for products and services is recognized. While theoretical knowledge regarding usability is described in a number of norms and standards, it is not much employed. This can be understood as an indication that the processes are not suitable for the SME target group yet. It is therefore not sufficient to build skills and expertise relevant to the user-centered software development. The conventional methods must be examined under the particular conditions for SMEs.

In the industry of information and communication the Federal Statistical Office Germany lists in total 130 311 enterprises. The majority of those enterprises has between 0 and 9 employees (120,140 companies) and belongs therefore to the micro enterprises. 7,823 companies employ between 10 and 49 employees. 1,953 companies employ between 50 and 249 people. Only 395 large-scale companies are registered in this branch. It is expected that the positive developments in the German software market in recent years will continue. According to forecasts business volume is expected to nearly double to 40.6 billion € (A.T. Kearney, 2008) by 2020. Extrapolated to the European Union and worldwide this paper assumes that there is a lot of money in this market. The target group of SMEs seems to be a very lucrative one.

5 USER-CENTRED DEVELOPMENT FOR SME's

Subsequently, the Integrated Method integrating usability into the mobile software engineering process is presented and explained. It provides a guideline for the professional usability engineering. It turns out that it needs to be adapted to the requirements of small-and medium-sized software companies. The procedure can be divided into two phases. In the first phase a generic requirements analysis is accomplished. The steps are based on the "Usability Engineering Lifecycle" by Mayhew (cf. Mayhew, 1999). The result of this phase is a first early prototype. To understand the basic operations a superficial process analysis is conducted (step 1).

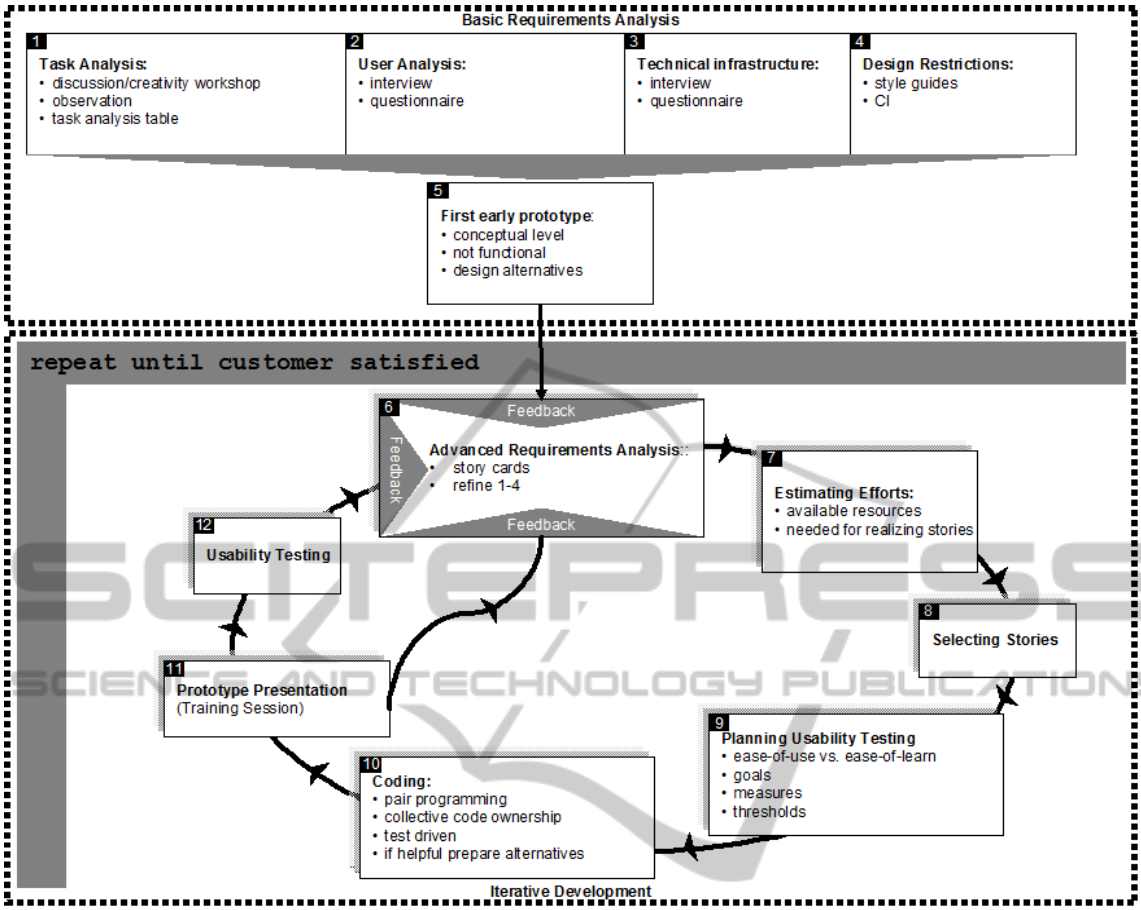

Figure 1: Integrated Method.

While doing so obvious problems are documented and people involved in the process are determined. For this well-known methods of process modeling may be used as well as the (contextual) Task Analysis Table proposed by Mayhew (see Mayhew, 1999).

As an extension with regards to mobility aspects the possible vicarious places and local types of the individual applications are collected. Common process notation can be enriched by mobile process landscapes proposed by Kuhn et al. (cf. Köhler and Gruhn, 2004). For an integration into BPMN see as well Damm (Damm te al., 2010).

Detailed user profiles are collected for persons and groups of persons who are affected by a future IT solution (step 2). In this step established methods of usability engineering are used as well (specific approaches for mobile applications can be found in Jones and Marsden, 2006 or Love, 2005).

In step 3 the technical infrastructure is examined. Finally, already existing design guidelines and standards will be collected and analyzed with regards to their applicability (step 4). Besides general design guidelines for mobile applications product-specific guidelines (e.g., Apple HIG Apple, 2011) can be consulted.

Based on the knowledge gained a first prototype solution will be developed (see also Casaday & Rainis, 1995, Beyer and Holtzblatt, 1999). This first prototype don't need to be fully digital and interactive (see e.g. Holzinger et al., 2011), but rather can be visualized in the form of drawings, etc. (for the use of prototypes for mobile applications, see Jones and Marsden, 2006; Love, 2005). These prototypes will be evaluated by users, e.g. in form of using scenarios.

If shortcomings in the conception emerge, those findings have to be continuously integrated as improvements in new prototypes. The described iterations are accomplished until the customer is satisfied with the outcome of the project or the project's budget is exhausted.

This is a dramatic reversal of the established treaty and implementing systems for software products. Those established systems even do not enable the use of agile software development

methods in SMEs. As an indication especially usability could measure customer satisfaction as an appropriate "termination criterion".

The customer feedback on the first prototype provides the input for the first iteration. Now the customer has a much better idea about the possibilities of mobile enterprise software than during the requirement analysis phase. With this knowledge the customer defines use cases (step 6).

The use cases are evaluated in particular with regard to their development efforts and are translated into task cards by the developer (see Wells, 2001) (step 7). Finally the customer selects the "stories" that can be implemented with the available resources during the upcoming iteration period (step 8).

Based on this selection the usability testing is being prepared. According to Mayhew (Mayhew, 1999) it will be initially decided whether priority is given to "ease-of-learning" or "ease-of-use". Then goals are provided, performance measures identified and finally thresholds determined. For example the goal of self-descriptiveness could be measured by the average error rate for a particular task.

As mentioned above, these metrics arise from experiences SMEs often are not able to access. The parameters and the description of use cases together with the derived test environment (context) are combined to a usability testing plan. Now the developer implement the selected use cases based on the usability requirements using the XP paradigms (see Wells, 2001) (step 10).

Here it is important to create transparency that usability criteria are used not only for approvals but also can be used as a development goal. Ideally the usability tests simulate the environmental conditions of the documented site for the according use case.

With the help of the usability tests the defined usability performance measures are collected and compared with the defined target values (step 12) in order to be able to comment the usability of the iteration results. The outcomes result in change requests and provide new use cases (again, step 6). Together with the in the current not implemented use cases this is the start for the next iteration.

6 DISCUSSION, CONCLUSIONS AND FUTURE WORK

The method of user-centered design of mobile enterprise software has been in successfully applied in numerous teaching and research projects by staff members and students. Regarding the application of the method and user-centered development in general in SME software projects problems occurred that have to be solved with future research in order to meet the demand identified in this paper:

- Experience is required for a more targeted use of the variety of methods. So far, knowledge is lacking what methods are suitable elements for SMEs and what not. Suitability depends on the SME and its business goals themselves as well as on project objectives, timeframe, skills and competition.
- The complexity of influences on usability aspects requires a broad range of expertise. A division of competence profiles on different does not seem feasible for SMEs. Usability engineering for SMEs will be streamlined by the proposed method. The method has to be defined in a way that the individual steps can be associated with clear job profiles and thus competence profiles (usability engineer, developer, interaction designer).
- Standardized and formalized processes can be important especially for occasional requirements engineering. Templates for surveys, personas, scenarios and use cases help to omit any requirements. This large number of artifacts of different media has to be managed centrally. But a corresponding "pragmatic" tool does not exist yet.
- The artifacts also serve to document the surveys. It is of particular importance that the knowledge of all parties involved (conceptual creator, designer, developer) can be retrieved. There is a lack of solutions as well.
- Besides various style guides the knowledge of good and bad approaches of interaction concepts is elusive. There are no central, easy to search collections of best practices or animated style guides for software developers available.

Those topics are addressed within a research project of the German Federal Ministry of Economics and Technology (BMWI).

REFERENCES

A. T. Kearney. (2008). *Die IT-Industrie im Jahre 2020*. Düsseldorf. Retrieved from http://tinyurl.com/ck6r8bz

Apple. (2011). *iOS Human Interface Guidelines*. Retrieved from http://tinyurl.com/32ra5kw

Baumeister, H. (2005, April). *AGILE: Architectures for Mobility*. Retrieved from http://tinyurl.com/d57phy2

Beck, K., & Andres, C. (2004). *Extreme Programming Explained: Embracing Change* (2nd ed.). Harlow: Pearson Professional Education.

Beyer, H., & Holtzblatt, K. (1999). Contextual design. *Magazine interactions*, *6*(1), 32–42.

Brau, H., Lehmann, A., Petrovic, K., and Schroeder, M. C. (Eds.) 2011. *Usability Professionals 2011: Tagungsband zur Konferenz Usability Professionals 2011 im Rahmen der Mensch und Computer.*

Burmester, M., & Görner, C. (2003). Das Wesen benutzerzentrierten Gestaltens. In J. Machate & M. Burmester (Eds.), *User Interface Tuning. Benutzungsschnittstellen menschlich gestalten.* Frankfurt am Main: Software und Support Verlag.

Casaday, G., and Rainis, C. (1995). Models, prototypes, and evaluations for HCI design. In J. Miller, R. Katz, R. Mack, & L. Marks (Eds.), *Human Factors in Computing Systems - CHI '95, Conference Companion. Mosaic of Creativity* (pp. 397–398). ACM Press.

Cockburn, A. (2005). *Crystal Clear: A human-powered methodology for small teams*. Boston: Addison-Wesley.

Cooper, A., Cronin, D., & Reimann, R. (2007). *About Face 3: The essentials of interaction design* (1st ed.). Indianapolis, IN: Wiley Publ Inc. Retrieved from http://www.worldcat.org/oclc/695567361

Damm, S., Ritz, T., & Strauch, J. (2010). Benutzerzentrierte Anforderungsanalyse für die Produktlinien-Entwicklung mobiler Unternehmens-software. In (Eds.): *LNI, Einflussfaktoren auf die Entwicklung flexibler, integrierter Informations systeme.* (pp. 117–130).

DIN EN ISO 9241-1 (2002-02-00): Beuth Verlag GmbH.

DIN EN ISO 9241-210 (2011-01-00): Beuth Verlag GmbH.

Ellerweg, R., & Ritz, T. (2008a). Benutzerzentrierte Entwicklung mobiler Unternehmenssoftware Teil 1. *net magazin*, (11).

Ellerweg, R., & Ritz, T. (2008b). Benutzerzentrierte Entwicklung mobiler Unternehmenssoftware Teil 2. *net magazin*, (12).

Hansmann, U., Merk, L., Nicklous, M. S., & Stober, T. *Pervasiv Computing* (2nd ed.). Berlin: Springer.

Hassenzahl, M. (2004). The Interplay of Beauty, Goodness, and Usability in Interactive Products. *Human-Computer Interaction*, *19*(4), 319–349. doi:10.1207/s15327051hci1904_2

Holzinger, A., Waclick, O., Kappe, F., Lenhart, S., Orasche, G., and Peischl, B. (2011). Rapid Prototyping on the Example of Software Development in Automotive Industry. In (Eds.), *Proceedings of the International Conference on eBusiness.* (pp. 57–61).

Jones, M., & Marsden, G. (2006). *Mobile interaction design*. Chichester, England ;, Hoboken, NJ: John Wiley & Sons.

Klauser, K., and Walker, V. (2007). It's about time: an affective and desirable alarm clock. In I. Koskinen & T. Keinonen (Eds.), *Proceedings of the 2007 Conference on Designing Pleasurable Products and Interfaces. 2007* (pp. 22–25). New York: ACM Press.

Köhler, A., & Gruhn, V. (2004). Mobile Process Landscaping am Beispiel von Vertriebsprozessen in der Assekuranz. In K. Pousttchi & K. Turowski (Eds.): *LNI, Mobile Economy: Transaktionen, Prozesse, Anwendungen und Dienste.* (pp. 12–24). Bonn: Ges. für Informatik.

Love, S. (2005). *Understanding mobile human-computer interaction*. Amsterdam [u.a.]: Elsevier.

Mayhew, D. J. (1999). *The usability engineering lifecycle: A practitioner's handbook for user interface design*. San Francisco, Calif: Morgan Kaufmann; London.

Norman, D. A., & Draper, S. W. (Eds.). (1986). *User Centered System Design*. Hillsdale: Lawrence Erlbaum Associates.

Ritz, T. (2003). Mobile CRM Systeme. *ZWF-Zeitschrift für wirtschaftlichen Fabrikbetrieb*, (12), 699–702.

Ritz, T. (2007). Die benutzerzentrierte Entwicklung mobiler Unternehmenssoftware. In B. König-Ries (Ed.), *Mobilität und mobile Informationssysteme. 2nd conference of GI-Fachgruppe MMS, March 6th, 2007 in Aachen, Germany.* Bonn: GI.

Roth, J. (2005). *Mobile Computing*. Heidelberg: dpunkt verlag.

Saleh, K., and El-Morr, C. (2003). *M-UML: extension to UML for the modeling of agent-based software systems*. Information and Software Technology 2003,

Salmre, I. (2005). *Writing Mobile Code*. Upper Saddle River NJ: Addison-Wesley.

Schwaber, K. (2004). *Agile project management with Scrum*. Redmond, Wash: Microsoft Press.

Wells, D. (2001). *Extreme Programming: A gentle introduction*. Retrieved from http://www.extreme programming.org/