10.20378/irb-58482

From Usability Workarounds to Usability Around Work

Kimmo Tarkkanen, Ville Harkke
Information Systems Science
Turku School of Economics, University of Turku
Rehtorinpellonkatu 3, 20500 Turku, Finland
kimmo.tarkkanen@utu.fi, ville.harkke@utu.fi

Abstract. In this paper, we introduce usability workarounds that take place in system design during IT tendering, and usability evaluation around work that characterizes an iterative IT product development. We claim that highly formal and detailed verification criteria and procedures in tendering cause usability workarounds by the IT vendors. To avoid workarounds and to bring usability activities more around work during IT tendering, we suggest applying open-ended usability test tasks for system validation and usability problem qualifiers as criteria.

1 Introduction

Many public information system purchases are subject to tender. Request-for-proposals (RFP) are documents that define a set of desired system requirements and a selection criterion for the proposed systems. RFP creation is a critical phase of governmental IS procurement also in terms of usability [1]. Jokela et al. [1] suggests that only user performance-based usability requirements can be verifiable, valid and comprehensive enough for RFPs. A performance requirement is e.g. how quickly the user should be able to accomplish a certain task. User performance can be measured in terms of effectiveness, efficiency and satisfaction i.e. the elements of the definition of usability [2].

We maintain, however, that applying very detailed performance requirements in the RFP may lead to distorted usability design at the IT vendor's site. HCI research has concentrated on the creation of RFP from the IT purchasing organizations' point of view, but how the usability practices of the vendors emerge and transform for tendering purposes are less studied. Here we present problems of usability design practices in a large and complex IT tendering case and then seek possible solutions and new openings from an agile, user-centred IT development case.

2 Two types of usability work

2.1 Usability workarounds

We participated in an IT tendering case as usability evaluation designers and researchers at the vendor's site [3,4]. We were responsible for measuring and improving the usability of the proposed IT system before it was introduced to the purchasing organization and exposed to their usability evaluation. The RFP contained formal user performance and satisfaction requirements and metrics as well as detailed information about how these were to be evaluated (e.g. usability tests tasks were published to the vendors). Thus, this allowed us to copy the evaluation practices (phase 1 in Figure 1) and prepare the system (phases 2-3) to best correspond with the requirements (phases 4-5). The usability criteria in the RFP included effectiveness as task completion rate (percentage), efficiency as number of errors made, time on task and number of interaction steps used, and satisfaction of users as measured with questionnaires (phase 0). These variables had nearly equal weights in overall system selection. A usability requirement was represented as, for example, a test task given to users to "*send an email message to John Smith*" (where the system could be a webmail). The buyer then evaluates how well their experienced email users perform the task with different webmail systems in usability tests (i.e. a common evaluation situation in system purchases for expert domains; phase 6).

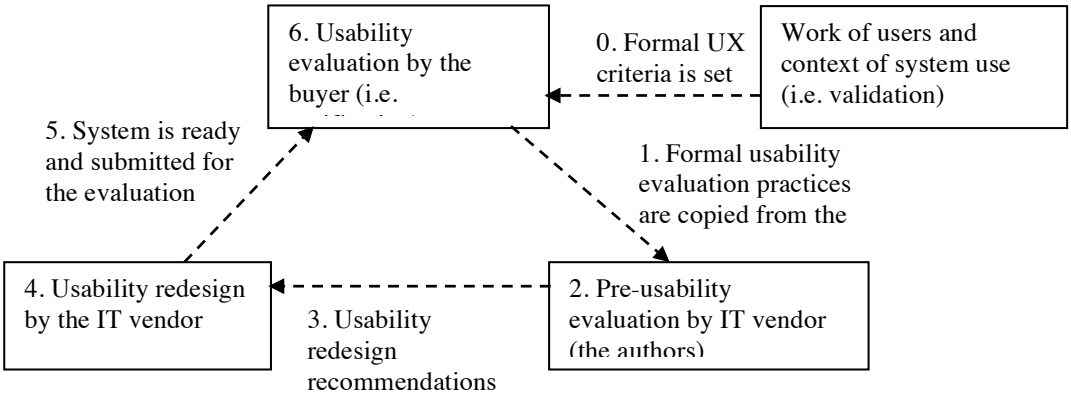

Figure 1. Usability evaluation setting in the tendering case.

The motivation and goal of vendors' usability work is success in tendering. The concept "workaround" presupposes that there is some sort of obstacle in achieving this goal that must be circumvented and hereby the 'standard

operating procedure' is bypassed or overridden, i.e. the work is carried out some other way than usually [5]. Thus, a usability workaround means bypassing human-centred design principles and usability guidelines and carrying out usability work some other way than usually due to confronted obstacles.

What are the obstacles the IT vendor may confront? First, usability design is limited by the time given and restrictions to modify the system. Second, the communication limitations with the buyer hinder usability design, because the vendor may want to know e.g. should John Smith be able to receive the message or who decides when the sending is completed (evaluators when send-button is pressed or when users verbally express?). Is the task completed successfully when a message is sent without content or title? These questions must have puzzled the buyer also, because usability requirements, procedures and criteria in the RFP were very detailed and rigid. Table 1 presents how certain characteristics of the RFP were confronted as obstacles by usability designers, implying usability workarounds and some further problems.

Table 1. Vendor's usability workarounds as responses to the RFP characteristics.

Characteristics of the RFP	**Usability workarounds**	**Problems**
Tendering schedule is tight	Perform only one evaluation-design iteration	Only minor and superficial changes can be implemented.
Usability test tasks are published	Design system usability to correspond the tasks	Task design should avoid giving clues to task completion.
Communication is limited	Interpret requirements to your system's advantage (when most efficient & effective)	If test tasks are subject to many interpretations, this may raise conflicts after the tendering.
Usability test tasks are low-level and strictly sequenced	No special workarounds. Design your system to support the requested task sequence.	The system may be more efficient and effective in another task sequence than requested.
Usability test tasks have interconnections and elements that recur	Analyse which test tasks are the most critical for other tasks to be completed efficiently and put design effort on the critical ones.	Design effort is misdirected from criticality in the real work.
Effectiveness is measured as task completion rate	Modify the terminology of the system (e.g. menu items, function names) and information appearance (dropdown lists) to correspond with the test tasks. Set also the critical views to be visible/open by default.	Renamed features, data ordering and visibility settings may serve only the test task, yet be inappropriate in other work tasks.

Efficiency is measured as execution time or as the number of navigation errors	Add shortcuts to shorten the navigation path despite of users getting lost in the system. Especially applicable when the next task begins from the main view. Inactivate features that are on the unrequested navigation path.	Users should be in control and informed about the system state and place (e.g. Nielsen's heuristics [6]).
User satisfaction is measured	Avoid showing error messages (e.g. by inactivation) not to give a feeling of erroneous system	In complex systems, error dialogs are inherent part of the systems.

The examples of different usability workarounds to improve task completion and performance efficiency were solely based on the RFP. Although such usability workarounds are possible for the system demonstration and testing purposes the resulting system configuration may not be feasible during the real use after the tendering. Therefore, formal UX definitions of the RFP must be based on an in-depth understanding about the current and desired work practices and needs of the users (i.e. validation). Definitely the buyer organization in this case possessed that knowledge. The main problem was the low-level and very detailed formulations of test tasks (i.e. descriptions users' work) and rigid requirements to comply with these. An implication was that the IT vendor focus was on usability verification (with usability workarounds) instead of on usability validation as a fit for purpose.

2.2 Usability around work

On the other extreme of the usability work, we partook in an agile development project of a mobile electronic patient record application for nurses in hospitals wards. This development was not yet subject to tender, but a traditional R&D activity of an IT company where such usability workarounds are not needed. Compared with the tendering situation, here was a closer user-developer relation and direct access to users during the usability work.

We conducted the first think-aloud usability tests of the paper prototype (although as pictures in a mobile tablet platform) in-the-wild with four nurses. Tests were preceded by an extensive user research by the application developers. Yet, and in contrast to the tendering case, our test approach was highly open-ended: The test task given to users was to '*arrive at workplace and begin to prepare your work shift with the [prototype] as a new application available to use*'. However, test administrators could pose additional questions ex tempore during the session. The motivation for the open-ended task was that a) users would express their needs more freely [7] and b) test would be based on real (yet simulated) tasks and, c) by not-restricting the view on the

application only (but covering the users' own articulation of their work) the test would challenge and help the continuous requirements elicitation process of the agile development.

Later, we categorized these findings based on usability problem qualifiers (i.e. failure qualifiers by [8]) (see also [9]). The "qualifier" attribute helps designers to see the real problem (i.e. what is wrong with the initial design) [8]. Ten categories were identified, which here point out that a single usability intervention – when it is practiced around work and freely in the collaborative environment unlike in the previous case – provides a broad spectrum of findings for usability verification and validation. Moreover, we want to draw attention to possible application of categories in tendering situation where the classification of usability problems could serve as a basis to set criteria in the RFPs.

Four out of ten categories represent poor fit in system-work relation: System functionalities or information elements are *missing*, *unnecessary*, *inadequate*, or *misplaced*. These are rather common findings from usability tests, yet for example missing functionalities and information items are hard to identify with classical, very low-level and only "doable" test tasks. Further categories include *misinterpreted* which represents problems in user-system relation i.e. users misinterpret the system (a single element of it or the whole purpose of it), *open design issues*, which are natural in the early development, representing problems found in developer-system relation and *technical deficiencies* which represent problems of system-specification relation (i.e. bugs/carelessness in implementation). Perhaps the most interesting findings and categories were those of developer-work relation. Some of the implemented functions were revealed to impose *problematic changes* in the work i.e. the system implementation will effect on current practices, existing social relations etc. in a way that users considered them possibly harmful and that designers' may have not understood the effect of some system features yet. In contrast, *unexplored design issues*, are "problems" that could impose positive effects and more value to work of users, if some more design and user research effort were put in the future.

Certainly there can be other categories too (e.g. category of positive findings [9]), as this analysis is based on a single case only. However, what makes these and similar categories together with open tasks feasible also in the tendering situation are that these bring us more profound understanding of what modifications need to be done to the system (and how much these may cost), in order to achieve a successful implementation. As buyers we could compare different systems in tendering against these categories. For example, we could count how many required features are missing, inadequately or

incorrectly designed, misinterpreted by the users or are imposing a problematic change. When each problem in each category is assigned to a feature in the system, we can assess the expected (or formally calculated) costs of implementation (versus benefits).

We do, of course, find the problems of system-work relation also when we test the system against the requirements of the RFP, because the RFP represents work. For example, a feature is missing, if it is requested in the RFP, but it is not implemented into a system. However, if the user fails to complete the task, it may be due to missing functions, missing information, inadequate implementations of the feature, misplaced features at certain point of task flow, or due to the features that are misinterpreted by the users. Thus, single user behaviour and measurement of it (task completion) may divide into several causes in the system (i.e. the problem categories). Similarly, efficiency as measured as time on task or number of errors may be due to many problems in different categories. If the systems are designed with usability workarounds for the verification test purposes only, the buyer may end up in situation where higher usability scores are given to the system that has more profound and costly problems. For example, two systems may be equally efficient in the task of sending email (i.e. having e.g. the same amount of interaction steps), yet the other system can miss a feature of subject field (not required by the task!), include an additional interaction step that is found unnecessary for the task at hand, have inadequate sorting and searching capabilities of email respondents, applies unfamiliar 'compose' and 'primary messages' (see www.gmail.com) instead of more familiar terms 'new' and 'inbox' (see MS Outlook) etc. With traditional measures, these differences would emerge only as a weaker user satisfaction and possibly without more detailed information about the type of the problem.

This is where the categorization of usability problem qualifiers differs from standard measures of usability: Categories do not only give scores but point to specific usability/utility issues in the system that are problematic and require improvements. As categories can have implicit relation to specific phases of human-centred design practices (understand, specify, produce), the buyer is able to evaluate the effort needed to fix the system. Features that are missing and denoting problematic changes at work require more comprehensive understanding of the context of use, while the requirements of misinterpreted and inadequately designed features are only re-specified and technical deficiencies are just re-implemented. The more fundamental the design flaw is and the more complex the future design need is, the more weight the problem category could have in the tendering. An evaluation based on problem categories presumably requires more effort by the buyer. However, we

maintain that systems' usability evaluation during tendering would be more comprehensive as it would contain a broader array of perspectives. Moreover, during the agile development project, categories could improve the design influence by steering and informing developers what features need what kind of modifications and what issues need more user research efforts.

The problem classification scheme is originally a result of applying open tasks in testing. When user-defined test tasks (e.g. open tasks) are applied in the system comparison, the buyer can ensure that, for example the missing system features are the ones needed in real work task accomplishment instead of making the comparison against the requirements in the RFP. This would bring the test closer to a validation situation where the usability requirements in the RFP do not need to be as comprehensive, detailed and rigid, but could be set in higher level (e.g. work roles, jobs, business processes to be supported). In order to test all the systems in a similar way by the specific user, the open task could be co-constructed in co-operation with the user (see [10]), after which it could be locked for this particular user. While problem categories help in answering what is wrong with the system from the viewpoint of users and work, categories cannot substitute usability criteria for time and safety critical systems where e.g. time on task is truly meaningful to measure. Open tasks would serve also better when the emphasis is on the overall system fit with the organizational tasks rather than only on collecting user-system performance data. More experimental research is required, in order to know how to best apply the problem categories and open tasks in the RFPs and IT tendering situations. Before that, we need more analytical research on category definitions as well as measuring and valuing them in the tendering process.

3 Conclusions

- Usability workarounds by the IT vendors may emerge in tendering when rigid and detailed usability requirements are set in request-for-proposals.
- Especially, low-level and detailed formulation of usability test tasks directed the usability design of the case towards sub-optimal solutions i.e. workarounds.
- High-level (usability test) tasks described in the RFP and open-ended usability test tasks for the verification and validation purposes around work could provide flexibility for IT vendors and confidence for IT buyers in introducing and selecting the best technology solution.

- Categories of usability problem qualifiers could be exploited as criteria in the system evaluation during the tendering, in order to bring the evaluation closer to future design and development needs of the system.

Acknowledgements

The corresponding author thanks MATTI doctoral programme of University of Turku for the travel grant.

References

[1] Jokela, T., Laine, J., and Nieminen, M. Usability in RFP's: The Current Practice and Outline for the Future. In Human-Computer Interaction. Applications and Services. Springer Berlin Heidelberg, 2013. pp.101-106.
[2] International Organization for Standardization. ISO 9241-11:1998 Guidance on usability. http://www.iso.org, 1998.
[3] Tarkkanen, K. and Harkke, V. Evaluation for Evaluation: Usability Work during Tendering Process. CHI'15 Extended Abstracts. ACM, 2015.
[4] Tarkkanen, K., Koskinen, J. J. and Harkke, V. Distorted Usability Design in Tendering. ETHICOMP'15, Leicester, UK. 2015 (Forthcoming)
[5] Alter, S. Theory of Workarounds. Communications of the Association for Information Systems 34. ACM, 2014. pp. 1041-1066.
[6] Nielsen, J. Usability Engineering. Academic Press, 1993.
[7] Tarkkanen, K., Reijonen, P., Tétard, F. and Harkke, V. Back to User-Centered Usability Testing. In Holzinger, A. et. al., eds. SouthCHI 2013, LNCS 7946, Springer-Verlag Berlin Heidelberg, 2013 pp. 91–106.
[8] Vilbergsdóttir, S. G., Hvannberg, E. T., and Law, E. L. C. Classification of usability problems (CUP) scheme: augmentation and exploitation. In Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles. ACM, 2006. pp. 281-290.
[9] Tarkkanen, K., Harkke, V. and Reijonen, P. Are We Testing Utility? Analysis of Usability Problem Types. In Proceedings of HCI International 2015, Los Angeles, US. 2015 (Forthcoming).
[10] Tarkkanen, K., Reijonen, P., Harkke, V. and Koski, J. Co-constructed Tasks for Web Usability Testing. In Proceedings of the International Conference on Interfaces and Human Computer Interaction. 22 – 24 July 2013, Prague, Czech Republic, IADIS Press, 2013. pp. 79-86.