Pers Ubiquit Comput (2006) 10: 345–355
DOI 10.1007/s00779-005-0050-7

ORIGINAL ARTICLE

Timo Jokela · Jussi Koivumaa · Jani Pirkola
Petri Salminen · Niina Kantola

Methods for quantitative usability requirements: a case study on the development of the user interface of a mobile phone

Received: 3 February 2005 / Accepted: 4 May 2005 / Published online: 8 October 2005

Abstract Quantitative usability requirements are a critical but challenging, and hence an often neglected aspect of a usability engineering process. A case study is described where quantitative usability requirements played a key role in the development of a new user interface of a mobile phone. Within the practical constraints of the project, existing methods for determining usability requirements and evaluating the extent to which these are met, could not be applied as such, therefore tailored methods had to be developed. These methods and their applications are discussed.

1 Introduction

Mobile phones have become a natural part of our everyday lives. Their user friendliness, termed usability, are increasingly in demand. Usability brings many benefits: users are able and willing to use the various features of the phone and the services supplied by the operators, the need for customer support decreases, and, above all, user satisfaction increases.

At the same time, designing is becoming increasingly challenging with the increasing number of functions and reduction of the size of the phones. Another challenge is the ever shortening life of the phones resulting in less time for development.

The practice of designing usable products is called usability engineering.[1] The book User-centered system design by Donald Norman and Stephen Draper [1] is a pioneering work. John Gould and his colleagues also worked with usability methodologies in the 1980s [2]. Dennis Wixon and Karen Holtzblatt at Digital Equipment developed Contextual Inquiry and later on Contextual Design [3]; Carroll and Mack [4] were also early contributors. Later, various UCD methodologies were proposed e.g. by [5–10]. The standard ISO 13407 [11] is a widely used general reference for usability engineering.

A general usability engineering life-cycle model — meant for various applications from information systems to personal devices such as mobile phones — is illustrated in Fig. 1.

The first activity is to identify users. Context of use analysis is about getting to know users: what the users' goals are in relation to the product under development, what kind of tasks they do and in which contexts. User information is the basis for usability requirements where the target levels of the usability of the product under development are determined. A new product should lead to more efficient user tasks; they are explicitly designed in the next step. Only at this stage does the user interface (UI) design start in the ideal case. All the information from the earlier phases forms the input to the interaction design phase. Interaction design and qualitative usability evaluations [2] typically form an iterative process (as does the whole usability engineering process). Various levels of prototypes are often used at this stage. When the design starts to mature, it is verified against the usability requirements.[3]

An essential part of the usability life-cycle is (quantitative) usability requirements, i.e. measurable usability targets for the interaction design [13–17]. As stated in

T. Jokela (✉) · N. Kantola
Oulu University, P.O. Box 3000, Oulu, Finland
E-mail: timo.jokela@oulu.fi
E-mail: niina.kantola@oulu.fi

J. Koivumaa · J. Pirkola
Nokia, P.O. Box 50, 90571 Oulu, Finland
E-mail: jussi.koivumaa@nokia.com
E-mail: jani.pirkola@nokia.com

P. Salminen
ValueFirst, Luuvantie 28, 02620 Espoo, Finland
E-mail: petri.salminen@valuefirst.fi

[1] Also other terms such as *user-centred design* and *human-centred design* are used, although some authors do not consider these terms synonyms.

[2] Often called formative usability evaluations.

[3] Often called summative usability evaluation.

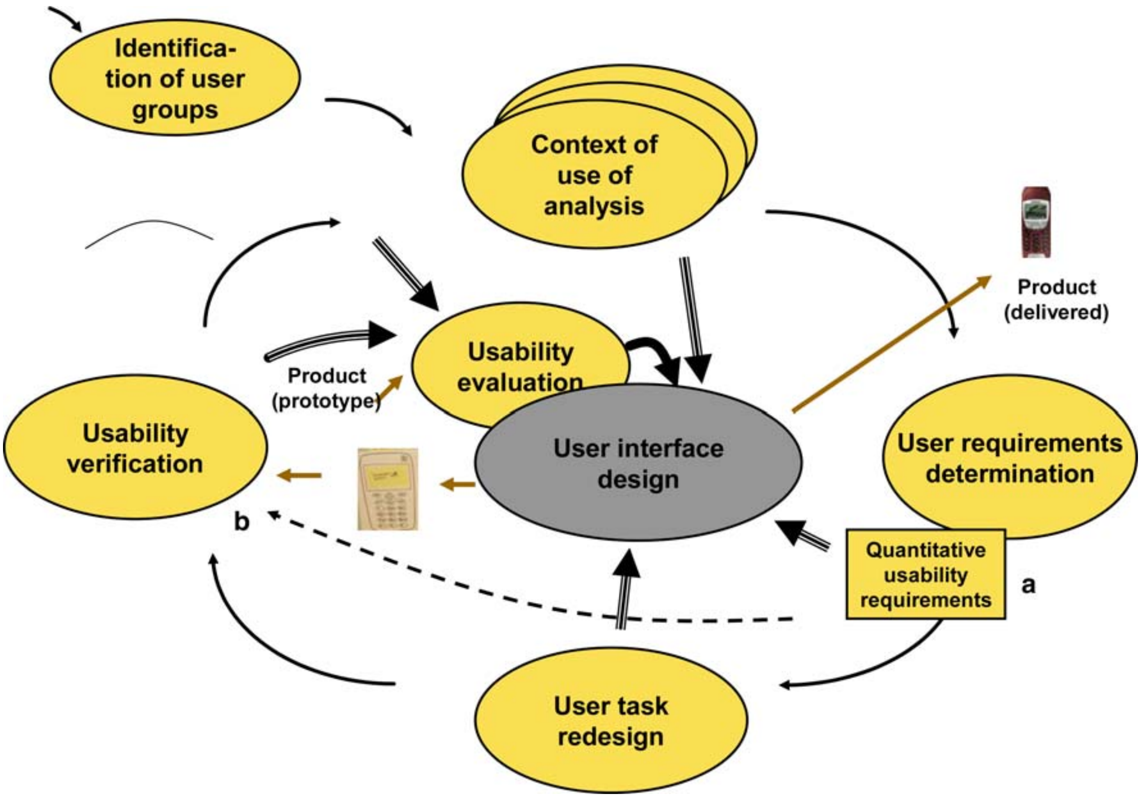
Identification of user groups
Context of use of analysis
Product (delivered)
Usability evaluation
Product (prototype)
Usability verification
User interface design
User requirements determination
Quantitative usability requirements
a
b
User task redesign

Fig. 1 Activities of usability engineering. Details of the life-cycle model can be found from [12]

[13]: "Without measurable usability specifications, there is no way to determine the usability needs of a product, or to measure whether or not the finished product fulfils those needs. If we cannot measure usability, we cannot have usability engineering".

The focus of this article is on the methods for quantitative usability requirements: methods for determining usability requirements (a in Fig. 1) and methods for evaluating compliance with the achievement of the requirements (b in Fig. 1) in the practical context of designing a UI for a mobile phone. We present a design case, in a practical context where it was neither feasible nor sensible to apply existing methods as described in the methods literature so we had to use non-standard methods.

A similar situation was faced by Wixon et al. at Microsoft who found it unfeasible to use standard usability evaluation methods and applied a non-standard method ('opportunistic usability evaluation') [18]. Wixon states that a key question of usability methods in a practical setting is, "what is the best way of deploying the usability resources we have available for this development cycle in order to maximize our beneficial impact on the product?" To find the answers, Wixon calls for case study research: "We need to evaluate methods in vivo, that is, by applying them to real products embedded in real engineering, corporate, and political environments and not to simulated systems or hypothetical models". Wixon finds that much of the literature on the evaluation of usability methods is "unhelpful, or even irrelevant" to the practitioner because the evaluations of the methods have been carried out in laboratory settings.

In this article, our aim is to meet the research challenge posed by Wixon: we present the methods that we used in a real development context of a mobile phone UI, for the determination of quantitative usability requirements and the evaluation of the compliance with them.

In the following section, a review of existing methods for usability requirements is provided. The case project is reviewed in Sect. 3. In Sect. 4, we describe the methods we used in determining quantitative usability requirements and for measuring compliance with them, our experiences in using them, and their advantages and shortcomings. In the last section, we summarise the results, discuss the limitations and present the implications for practice and research.

2 Methods for quantitative usability requirements

There are two main activities related to quantitative usability requirements. During the early phases of a development project, the usability requirements are determined (a in Fig. 1), and during the late phases, the usability of the product is evaluated against the requirements (b in Fig. 1). Determining usability requirements can be further split into two activities:

defining the usability attributes, and setting target values for the attributes. In the evaluation, a measuring instrument is required. In this section, a review of methods supporting these activities is given.

2.1 Determining usability attributes

The main reference of usability is probably the definition of usability in ISO 9241-11: "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" [19]. In brief, the definition means that usability requirements are based on measures of users performing tasks with the product to be developed.

– An example of an effectiveness measure is the percentage of users who can successfully complete a task.
– Efficiency can be measured by the mean time needed to successfully complete a task.
– User satisfaction can be measured with a questionnaire.

Usability requirements may include separate definitions of the target level (e.g. 90% of users can successfully complete a task) and the minimum acceptable level (e.g. 80% of users can successfully complete a task) [20]. Whiteside et al. [21] suggest that quantitative usability requirements be phrased at four levels: worst, planned, best and current.

Questionnaires measuring user satisfaction provide quantitative, though subjective usability metrics for related usability attributes. For example, with the SUMI [22] questionnaire it is possible to measure five usability attributes: affect, efficiency, helpfulness, control and learnability. The system usability scale can be obtained through the SUS [23] questionnaire and QUIS [24] covers attributes such as readability of characters, layout of displays and terminology.

2.2 Methods for determining usability targets

Usability engineering literature agrees that determining quantitative usability requirements should be a collaborative effort. However, there are no very concrete guidelines as to how this effort should be organized and managed. The existing literature mainly focuses on describing and exploring the concepts and formats related to the definition of usability and the contents of the usability requirements document [25].

Possibly one of the most detailed guidelines for determining usability requirements is a six-step process by Wixon and Wilson [14]. In their process, relevant usability attributes are determined based on user profile and task analysis. Then the measuring instruments and measures are decided upon and a performance level is set for each attribute. They agree with Whiteside et al. [21] that four performance levels can be set for each attribute and determining the current level lays the foundation for setting other levels. If the product is new, measurements for the current level can be attained, for example, from an existing manual system. Like Hix and Hartson [6], Wixon and Wilson [14] suggest that in the beginning two to three clear goals that focus on important and frequent tasks are enough and later, as the development teams accept the value of usability goals, more complex specifications can be generated.

Gould and Lewis [26] state that developing behavioural goals must cover at least three points. Firstly, a description of the intended users must be given and the experimental participants should be agreed upon. Secondly, the tasks to be performed and the circumstances in which they should be performed must be given. The third point of the process is giving the measurement of interest, such as learning time and the criterion values to be achieved for each. According to them [26] behavioural criteria, for example learning time, is usually relatively easy to specify, but iteration is needed when defining the appropriate values for these criteria. In the case of new systems, iteration is required even to identify the criteria correctly.

According to Nielsen [5] usability is associated with five attributes: learnability, efficiency, memorability, error and satisfaction. In usability goal setting, these attributes must be prioritised based on user and task analysis, and then operationalised and expressed in measurable ways. Nielsen states that goals are relatively easy to set for new versions of an existing system or systems that have competitors on the market. In the case of completely new systems he proposes an approach, where a set of sample tasks are defined and then several usability specialists are asked how long it should take users to perform them.

Dumas and Redish [27] suggest that when planning for performance measures, it is important to consider where the product is in the development cycle. For example when testing a prototype, performance measures such as repeated error and frustration make sense, but time probably does not. They suggest that in setting the criteria for these measures, information about the job and task analysis will help. Furthermore, it is an advantage to have an expert doing the task.

Mayhew [7] introduces a nine-step procedure for setting usability goals. In her procedure qualitative usability goals are first identified and prioritized. Then those qualitative usability goals that are relatively high priority and seem easily quantifiable should be formulated to quantified goals. She suggests that narrow feature-oriented usability goals will be appropriate while developing a new version of an existing product and broad task-oriented goals will be most useful while developing a completely new product.

There have also been various projects where enhanced usability engineering methods have been developed. For example the MUSiC methodology [28] aims to provide a comprehensive approach to the measurement of usability. It includes methods for specifying and measuring usability during design. One of the methods is the performance measurement method, which aims to provide a means of measuring two of the ISO 9241-11 standard usability components, i.e. effectiveness and efficiency. The method includes a sequence of steps, which are guided and supported with the MUSiC tools such as handbooks and software [29]. ISO 9241-11 [19] itself states that usability measures can be specified, "focusing ... on the most important user goals may mean ignoring many functions, but is likely to be the most practical approach".

Also the RESPECT project has produced a number of usability documents including a set of handbooks on usability design. One handbook, User-centred requirements handbook [30], is concerned with user-centred requirements specification. Its aim is to provide a structured basis for gathering user requirements equivalent to the specification of business requirements and technical requirements. However, the handbook is mainly concerned with qualitative usability requirements but does not discuss quantitative requirements in detail.

Furthermore, there are very few empirical research reports on quantitative usability requirement methods in practice. One of the few reports is by Bevan et al. [31] who conducted case studies on quantitative usability evaluations following the Common Industry Format for usability testing, CIF [32] in a purchaser-supplier setting. Two of the case studies also included the determination of quantitative usability requirements. In one case, five experts and four non-experts attempted to complete different sets of "typical tasks" in an existing system, and the goal of the new system was to "at least equal and if possible improve on these success rates". Another case included a step "specification of the usability requirements". It is reported that the cases were successful; however, the methodological aspects are not discussed in detail.

2.3 Methods for quantitative evaluation of usability

Whether the quantitative requirements have been met can be determind through a usability test. Wixon et al. [14] define the term "test" as a broad term that encompasses any method for assessing whether goals have been achieved, like a formal laboratory test or a collection of satisfaction data through survey. However, they point out that, although quantitative data can be gathered in many settings, usability engineering does imply that a systematic approach has been taken. Tasks must also be standardized and the influence of extraneous variables minimized.

When evaluating usability, ISO 9241-11 [19] claims it is important that the context selected be representative. Evaluations can be done in the field in a real work situation or in laboratory settings in which the relevant aspects of the context of use are re-created in a representative and controlled way. A method that includes representative users performing typical, representative tasks is generally called usability testing.

Tasks that are done in usability testing provide an objective metric for the related usability attribute. Hix and Hartson [6] indicate that tasks must be very specifically worded in order to be the same for each participant. Tasks must also be specific, so that participants do not get sidetracked into irrelevant details during testing. Wixon et al. [14] suggest that during the test, the tester should minimize the interaction with participants. Butler [33] describes his approach where "seven test users were given an introductory level problem to solve, then left alone with a copy of the user's guide and a 3270 terminal logged onto the system."

User preference questionnaires provide a subjective metric for the related usability attribute such as ease of use or usefulness. Questionnaires are commonly built using Likert and semantic differential scales and are intended for use in various circumstances [32]. There are a number of questionnaires available for quantitative usability evaluation, like SUMI [22], QUIS [24] and SUS [23]. Karat [34] states that questionnaires provide an easy and inexpensive method for obtaining measurement data on a system. However, they also have shortcomings, for example information may be lost by the limited nature of the questions.

Usability can be quantitatively evaluated also with theory-based approaches such as GOMS and keystroke level model, KLM [35]. With GOMS, for example, total times can be predicted by associating times with each operator. According to John [36], GOMS can also be used to predict how long it will take to learn a certain task. With these quantitative predictions GOMS can be applied for example in a comparison between two systems. The GOMS model also has its limitations. Preece et al. [37] suggest that GOMS can only really model computer-based tasks that involve a small set of highly routine data-entry type inputs. The model is not appropriate if errors occur.

KLM is a simplified version of GOMS. It provides numerical predictions of user performance which are developed based on extensive empirical studies by Card et al. [35]. The model is comprised of several operators such as keystroke, point and a generic mental operator. There are also heuristic rules, which are provided to predict where the mental operators are needed.

There are also simplifications of the KLM model, such as the one which considers only keystrokes, and the execution time is proportional to the number of keystrokes. These simplifications are less accurate in predicting execution time, but they do provide the designer with a greater ease of use [35].

3 Review of the case project

Our case study is a past development project of a new UI concept for a mobile phone at Nokia. The UI concept includes the design of all the main elements of the UI apart from software implementation: input (selection of keys) and output (display) devices; industrial and graphical design; and dialogue and interaction design principles.

The project had a clear business driver: the usability of the internet access (specifically, WAP[4]) functions had to be at a clearly of a better level compared with the old UI. The project had a tight schedule. It also had other constraints, one being that the usability of the UI had to be quantitatively evaluated very early in the development life cycle.

3.1 On the design context and challenge

Cellular phones are mechanical devices. While they are manufactured in large quantities, there are very high requirements for the quality of the mechanical design and components. In order to provide time for industrial and mechanical design, one has to make a decision on the mechanical elements very early in the development life cycle, including two elementary components of UI: the set of keys and the type of display. In particular, the early decision about the keys makes a difference in the development process compared to desktop software systems. A design issue that comes typically at the detailed design phase of a desktop UI (choosing the push buttons) needs to be decided very early in the development of cellular phones.

The company practice is to use quantitative targets in R&D projects, to orient a development project in the right direction and to make objective evaluation of the project possible. Qualitative usability targets were used in an earlier UI concept development project, but the experience was not very positive as it proved very difficult to evaluate at the end of the project the extent to which good usability was truly achieved. Therefore, the management required quantitative usability targets this time. While the usability targets could not be the only ones in the project, the total number of usability targets was limited to maximum four.

Early decision making on the key UI elements based on quantitative usability evaluation set special requirements for our project. Because the set of keys cannot be designed in isolation from the other parts of the interface, the practical requirement was to design all the main elements of the UI equally early. We anticipated that no working prototypes or the UI software would be available at the time of the evaluation. We further anticipated we probably would not even have time to build interactive simulations that run on a computer. Thus, we had to be able to quantitatively evaluate the usability of the UI concept at a phase in the development where no extensive usability tests, for example, could be made in a traditional way.

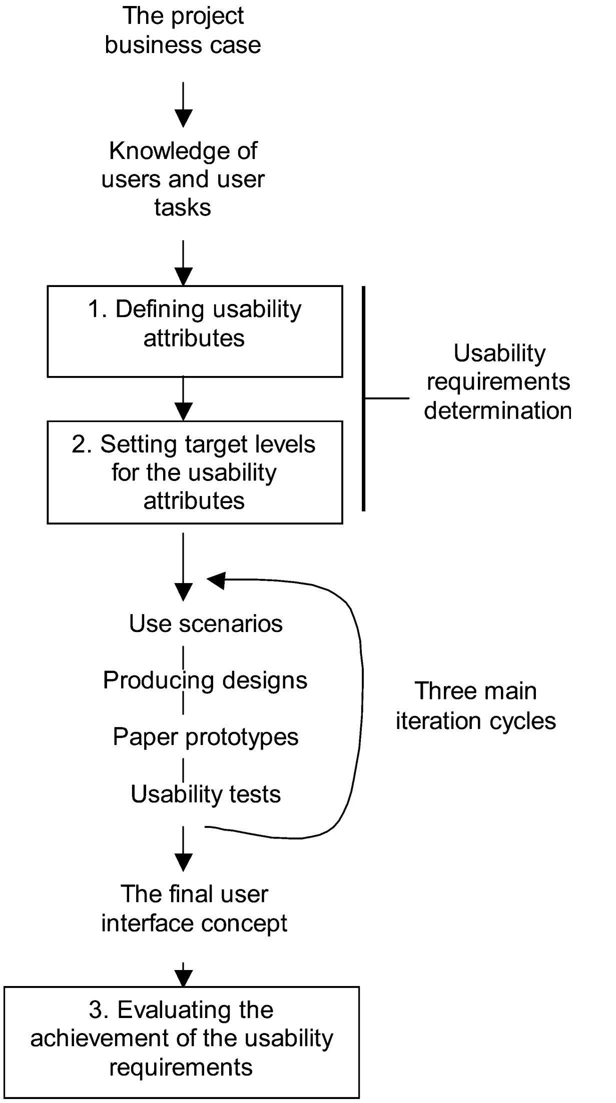
The project business case
Knowledge of users and user tasks
1. Defining usability attributes
2. Setting target levels for the usability attributes
Usability requirements determination
Use scenarios
Producing designs
Paper prototypes
Usability tests
Three main iteration cycles
The final user interface concept
3. Evaluating the achievement of the usability requirements

Fig. 2 The flow of the project. The activities in the boxes relate to usability requirements

3.2 The flow of the project

The overall lifecycle of the project is illustrated in Fig. 2. As background information, we had knowledge of the users and their tasks. The first step was to define the usability attributes and determine quantitative usability targets. Other measurable targets for the project, such as development time, were also agreed at this point.

As mentioned, the demand for setting quantitative usability requirements came from the management. The usability practitioners of the project team first opposed the idea; the task was perceived as too challenging. After considerable brainstorming and other efforts we ended up with the approach described in this article. Determining the quantitative usability requirements consisted of two separate tasks: (1) defining an appropriate set of usability attributes, and (2) determining target levels for these attributes.

[4] Wireless Application Protocol.

An iterative design cycle followed:

- Use scenarios were produced on the different applications of a phone. These scenarios were later used as the basis for paper prototype design and usability tests;
- hypothetical UI concept was produced. In the beginning, this was based on our assumptions; in the later rounds of iteration, it became more and more based on the data from the user tests;
- Paper prototypes [38] of the UI were produced in workshops in which the key members of the project team, especially UI designers, participated. Design tasks were assigned to each participant (i.e. typically one application per person);
- Usability tests were conducted on the paper prototypes.The project had a tight schedule. We had time for three main iteration cycles (of the steps described above) where different UI designs were explored with different phone applications. Different concepts were identified, and qualitative usability evaluations were carried out through paper prototyping. We perceived the usability requirements of the project as challenging; several different UI concepts had to be explored before we felt that we had a good design solution.

Finally, after the last iteration cycle, the final design (3) was verified against the usability requirements to check whether the right design was selected and to determine the extent to which the design meets the predetermined usability targets. Even if the usability targets of the project were well known by the project team, there was some anticipation in the team about the outcome. The results, in turn, clearly indicated that the project had been successful: all the usability targets were reached and even exceeded. The results were reported to the management board, who approved the results and the project staff received their project bonuses accordingly. A special reward was that the project was selected as the best R&D project in the all Nokia quality competition of that year.

4 Methods related to usability requirements

In this section, we first discuss the definition of usability attributes and the target setting. Then we describe the instruments for measuring the attributes, and how the evaluation was carried out. At the end, we summarise the advantages and shortcomings of the methods.

Table 1 Example of user tasks on basic phone functions

Calling by entering digits manually
Calling a person from the phone book
Using quick dialling to call
Inserting a new entry in the phone book
Editing names or numbers in the phone book
Answering a call
Answering a waiting call
Switching between two calls
Rejecting a call

4.1 The usability attributes

General usability attributes include usefulness, learnability, efficiency, error rates, memorability, etc. These should be transferred to specific usability attributes that “reflect the user’s work context” [14]. There were a number of factors in our project that had to be considered when selecting the specific usability attributes:

- As discussed above, a main restriction was that the total number of usability goals was limited to four measures.
- A large number of end user tasks are characteristic of cellular phones. One can identify many tasks related to the basic phone functions only, as illustrated in Table 1. When we take into account all the other applications of a typical cellular phone — e.g. text messaging, different settings (ringing tone, volume) and modes of operation, call transfer and forwarding functions, calendar, voice messaging, new data applications — the number of user tasks is very high. In addition, WAP — the specific driver for the development of this UI concept — meant another set of new user tasks. The high number of time-critical tasks meant that determining usability requirements based on a small set of tasks did not make sense: a UI should be designed so that all the time critical tasks could be completed efficiently.
- Cellular phones are consumer products that should be easy to learn, even without a user manual. But another characteristic of cellular phones is that they are products that are used frequently, many times a day. This means that efficiency — everyday use — is also a critical quality attribute. This aspect had to be taken into account when determining the usability attributes.
- Further, we had to be able to measure the achievement of the usability attributes in the context of the project. We had to be able to have the usability measured in a situation where we do not have a functioning prototype of the product. The attributes had to be defined so that measurement at this stage was feasible.
- Last, but not least, the usability requirements had to be relevant for the business case. They had to direct the design of the UI in line with the business goals. In this case, the key driver was to improve the usability of the WAP functionality. On the other hand, it was required that the design would not compromise on the usability of the other phone functions and applications.To meet these requirements, our first decision was to use *relative* usability attributes, i.e. to compare the usability of the UI under development with a reference interface. This practice is adopted e.g. in [31] where it is proposed that “usability for the new system should be at least as good as for the old sys-

tem". Probably a more common practice is to use absolute attributes, such as performance time, or number of errors. We anticipated that relative usability attributes would make the target setting easier as it was not necessary to have any previous benchmark data, and also the measurement will become more practical. An earlier Nokia UI was chosen as the reference.

We ended up defining the usability requirements with two attributes: *relative average efficiency*, and *relative overall usability*. Both these attributes are adoptions of our own from general usability attributes. With these two different attributes, we aimed to get a balanced set of the usability requirements that would depict the true usability of a phone UI.

4.1.1 Relative average efficiency

Efficiency is a relevant usability attribute of a mobile phone UI because it relates to the daily use — 'power use' — of the product, after a user has learnt how to use a product. An efficient UI concept makes it possible for a user to perform the most important tasks quickly.

Traditionally, efficiency is defined based on individual user tasks, i.e. by defining the time that a user takes to complete tasks. This approach, however, was not logical in our context because we were limited to a maximum of four usability measures. The selection of only four tasks had been too small a sample of time-critical user tasks of a mobile phone.

Our solution was to use an attribute that we call *relative average efficiency, RAE*. It is an objective user task based attribute, based on the following steps:

- Select a reference UI (for example, the UI of an existing product).
- Identify the time critical user tasks.
- For each task, determine the relative efficiency of the new UI compared with the reference UI.
- The relative average efficiency is the average of the relative efficiencies of individual tasks, expressed in percentages.For example, the value of RAE '35%' means that the new user interface is 35% 'more efficient on average' than the reference one. The value '−10%' means that the new user interface is 10% 'less efficient on average' than the reference one.

4.1.2 Relative overall usability

To complement the efficiency aspect, we used a general attribute *relative overall usability (ROU)*. The overall usability is meant to cover the general design principles of the UI: how quickly users learn to use the phone; whether the design metaphors and paradigms are sensible and the use of keys is logical; whether the style remains consistent through the different applications and users do not make errors (and if they do, they can recover from them) etc.

Table 2 The usability attributes (x) of the case project

Applications	Relative average efficiency	Relative overall usability
WAP	X	X
Phone functions	X	X

ROU is based on the following concepts:

- Evaluation is based on the judgement of usability experts.
- The experts are asked to give one rating both for the new and the old UI with the Likert type scale.
- The relative overall usability is the average difference between the ratings given to the new and the reference UI.

4.2 Determining target values for the usability attributes

After a number of brainstorming sessions and discussions with the management, we decided to determine the usability targets separately for the traditional phone functions and for the WAP functions. As a result, we had four attributes (Table 2). This separation was deemed sensible because the key business driver was a clear improvement on the WAP usability. We wanted to set tougher usability requirements for the WAP usability than for the usability of the phone functions.

The challenge was to determine what would be the appropriate target levels for the attributes. Setting the exact numeric values for the usability attributes was more guesswork than science. The goals were set together with the management ambitiously, realistically and in line with the business aims. Moreover, it was the first time this kind of goal setting was done in the company.

It was agreed that reasonable targets in the WAP functions were a 15% increase in RAE, and 1 point increase in ROU. We anticipated that a UI designed for WAP might easily lead to compromises in the usability of phone functions. Therefore, it was agreed that a reasonable target was to keep the usability of the phone functions at the same level as in the reference UI. In summary, we ended up with the targets presented in Table 3.

4.3 Measuring instruments

Measuring instruments are needed to evaluate in practice the compliance with the usability requirements. We had a number of limitations in selecting the measuring instruments. At the time when the usability had to be measured, we did not have a working prototype of the new UI. The design challenge had been so demanding that we needed to explore several design ideas on paper prototypes, but did not have time to implement a prototype of our final design. We

Table 3 The target levels of the usability attributes (in relation to the reference UI)

Applications	Relative average efficiency	Relative overall usability
WAP	Increase of +15% compared with the reference	Increase of 1 point (on a scale of 1...7) compared with the reference
Phone functions	0%, i.e. the same level as the reference	0, i.e. the same level as the reference

did not aim for a thorough specification and documentation of the UI in the short period of time that we had. What we had was a set of scenarios on how the new UI would behave in different situations and with different applications, documented in storyboards. In addition, the team had the details of the design in their minds: how our design would work in different applications and situations. The easy part was that the reference user interface was available to the evaluators— an existing Nokia phone.

4.3.1 Instruments for measuring RAE

At the time of setting the target levels, we did not specify what exactly an ‘increase of 15% in efficiency’ means. To make the measurement of relative average efficiency practical, we chose a strategy whereby we calculated how many end user tasks would be more efficient with the new UI. In other words, we did not calculate the absolute times (or time estimates) for the tasks. We ended up with two formulas, *relative average task efficiency measure(RATEM*) and *simplistic keystroke level model(S-KLM)*.

RATEM is for calculating the relative average task efficiency when the relative individual task efficiencies are known, Equation 1.

Equation 1. RATEM the formula for calculating the relative average efficiency increase in percentages

1. Determine a representative set of efficiency critical tasks.
2. For each task:
   - If the new concept is more efficient than the reference, the score = 1.
   - If the concepts are equal, then the score = 0.
   - If the new concept is worse, the score = −1.
3. The improvement in the relative average efficiency = (sum of scores/number of tasks) × 100%

For the practical estimation of the efficiency of the individual tasks, we developed *S-KLM*, a simple version of the keystroke level model, KLM (Card 1983). Since it is enough to know which UI concept is better, we did not need to make exact estimates of the times for performing a task. Our simple formula for calculating the task efficiency is illustrated in Table 4:

For example, if a user task takes only one key press, the score is 1. If a user task is one long press, the score is 1.25. If a user task takes two presses of keys not close to each other, the score is 2.

4.3.2 Instrument for measuring ROU

The ROU was measured through expert evaluation. Three usability experts (in one case: a team of experts) would evaluate the concepts independently based on their professional judgements, without mutual communication. The evaluators were given the freedom to select an evaluation method that they found appropriate. The evaluators were asked to score both the new UI and the reference UI on a Likert scale 1 (poor, would not recommend it to anybody) ... 7 (excellent, would strongly recommend it to a colleague).

The final score, ROU, is the average of the individual scores, Equation 2.

Relative overall usability measure (ROUM) a formula for calculating the relative overall usability increase

- ROU increase = sum of scores (U new UI − U old UI)/ $n \times 100$ %
- Where U new UI = the overall usability rating of the new UI
- U old UI = the overall usability rating of the old UI
- n = number of evaluators

In addition, it was presumed that the evaluators would also give qualitative comments. The reason was to gain confidence in the validity of the evaluation. If the comments were in line with each other, it would confirm the validity of the results.

4.4 Implementation and the results of the evaluation

The evaluation had to be carried out with the limitations discussed above: we had only limited documentation of the user interface, but the team had clear ideas how the user interface would work in different situations. Our strategy was to organise an evaluation session where the design team was at the disposal of the evaluators. The team presented the features of the user interface by making presentations, answering questions, etc.

Table 4 The simplistic key-stroke level method, S-KLM

First button press	= 1
A long press	= + 0,25
A second press	= + 0,50
Pressing an adjust key	= + 0,75
Pressing any other key	= +1

4.4.1 *Measuring the RAE*

We first identified the efficiency critical user tasks. This was done together with representatives of product marketing and application specialists, based on the existing user data. The main criteria for the selection of the efficiency critical tasks were (1) tasks that are done frequently, and (2) tasks that need to be done quickly for other reasons. As a result, we ended up with 20 ... 30 user tasks for both the phone and WAP functions.

A usability expert (not belonging to the project) made evaluation, asking details from the key designers of the project when required. The evaluator produced a table that showed all the tasks, the relative efficiency rating of each task and the RAE result, Table 5.

4.4.2 *Measuring the ROU*

Three usability experts (in one case: a team of experts) evaluated the ROU separately for the WAP functionality and phone applications. We organised a presentation session where the new UI was presented to three evaluators. We briefed the evaluators about our business case: i.e. what the drivers were, what the target user segment was, and what the key applications were. While we did not have a running prototype, we presented the user interface on slides and answered the evaluators' questions. The evaluators were asked to send the results (ratings and qualitative justifications) to the project team soon after the presentation session.

The overall usability results were quite consistent even though the evaluators did the work independently without any mutual communication. An interesting aspect was that they used different usability evaluation methods: two teams used heuristic evaluation [39], while one team used SUS [23]. In addition, the qualitative feedback from the evaluators was quite consistent. This made us believe that the results of the 'ROU' were quite valid.

4.5 Advantages and disadvantages of the methods

In summary, our approach includes the following new methods related to usability requirements:

- Two specific usability attributes: RAE, and ROU.
- Two instruments for measuring the RAE: RATEM and S-KLM.
- An instrument for measuring the ROU: ROUM.

REA and ROU The chief advantages of these usability attributes are that usability can be expressed with few numbers only, and the attributes basically take into account widely different aspects of usability. RAE measures efficiency based on several user tasks, and ROU measures other usability aspects (learnability, errors, etc.). Probably the main disadvantage of the attributes is that they may hide individual usability flaws: an average rating may be quite good but still there may exist even significant individual usability problems in the user interface.

RATEM and S-KLM The advantage of these measuring instruments is that the usability can be measured efficiently — with few resources only — and with lo-fi prototypes. Measuring the relative efficiency of individual tasks with S-KLM is rather straightforward: it proved easy to determine which one of the two UIs provided a more efficient way of accomplishing a task. A detailed level keystroke level analysis, for example, would take considerably more resources. Calculation of the RAE result with RATEM is also very easy as it involves a simple formula or table.

A disadvantage of the methods is that the results do not reveal the efficiency in terms of absolute time. The results show how large portion of the tasks are more efficient. For example, S-KLM gives a score '1', no matter whether a user task is only marginally or significantly more efficient to perform with the new interface. The RAE rating may be quite high even if the absolute efficiency improvement is not much. This phenomenon was actually evident in our case: the evaluation results proved to be very good (i.e. we well exceeded our targets). Another, natural weakness is that the evaluation is done without users. The paths for carrying out the tasks are decided by the evaluator and designers, not by the actual end users.

ROUM The advantage of this method is that it is, as is RATEM and S-KLM, practical for the design team. Further, the independent evaluators with different methods complement each others' results. A weakness of the method is that of all expert methods: the results are very much dependent on the evaluators, not based on true end user data.

Summary The strong points of our methods are that they are efficient, they can be used with lo-fi prototypes, and they guide the development team to think about 'the broad picture' of usability instead of, for example,

Table 5 The calculation of the RAE (an example table)

Task	Ratings		
	The new UI more efficient	Equal	The reference UI more efficient
Task 1	x		
Task 2		X	
Task 3	x		
Task 4	x		
Task 5 Etc.			x
Summary	16	8	6
RAE	(16 − 6)/30 × 100% = 33%		

focusing on some individual tasks only. The main weakness is the potential validity problems; especially as the instruments do not include end user participation and the results may hide even significant individual usability flaws.

5 Discussion

A case study is described where quantitative usability requirements were determined for a new UI of a mobile phone, and later the compliance with the requirements was measured. The case threw up challenges — due to the specific characteristics of mobile phones (e.g. the need for early decisions on keys) — which made it inappropriate to use standard methods. New methods for usability requirements determination and evaluation had to be developed within the practical constraints of the project. The methods are described, the rationale behind the methods is given, and the advantages and disadvantages of the methods are discussed. Although the user interface concept was never fully implemented [5] and therefore its usability could not be reliably evaluated, we were confident that we gained good results in designing high-level usability.

The case study contributes to show how it is possible to tailor usability methods to the true context of real development projects. Our case concerned the development of a mobile phone user interface. However, the results and findings of this research should be applicable to other kinds of development projects, too. Quantitative usability requirements are useful regardless of the type of the product.

We are quite conscious of the limitation of the methods we used, as discussed in the previous section. The methods described in this article should be understood as examples of how quantitative usability requirements can be determined and evaluated in practice in the context of one particular case project, but not as 'the' methods for general application.

5.1 Implications to the practice

In conclusion, we recommend the use of quantitative usability requirements in development projects. Such requirements make usability a true issue in the project and give a vision for design. However, a practitioner should take the following issues into account:

- To be effective, usability requirements should not be employed only because they are part of a usability engineering methodology. They should be defined as true project targets to be monitored by the management. Usability should be among the criteria adopted to evaluate the success of the project.
- Determining appropriate usability attributes and setting target values for them is a challenging task. Usability requirements should be defined so that they depict a 'usable product' as well as possible. The difference between some usability attribute and a descriptive set of usability attributes should be understood. In our case study, we defined usability attributes to include both efficiency and general usability aspects. It should be understood, however, that the appropriate set of attributes is heavily dependent on the product or application.
- The determination of quantitative usability requirements and their evaluation should be distinguished. We propose that it is not necessary to know how to measure them exactly at the time of determining the requirements. An important role of usability requirements is that they give direction and vision to the user interface design. This experience is shared by Wixon et al. [14]: "even if you do not test at all, designing with a clearly stated usability goal is preferable to designing toward a generic goal of 'easy and intuitive'".
- We encourage innovativeness in usability methods. It is seldom possible to use usability methods ideally. This article presents our innovations on the methods for determining and evaluating usability requirements. An appropriate approach in another kind of project context would most probably not be the same as the one described in this article. The project context and the business case always have a major impact on the usability attributes.

5.2 Topics for future research

We find that too little empirical research has been done on the topic of measurable usability requirements. The literature mainly focuses on presenting the formats of usability requirements. There are very few practical guidelines on how to derive the 'right set' of usability requirements; i.e. how to define an appropriate set of usability attributes and how to set appropriate targets for the attributes. There is much more literature available on methods for evaluating usability. On the other hand, methods for quantitative usability evaluations tend to be rather resource intensive. There is thus a need to develop practical methods applicable in real development situations.

6 Conclusion

We described a case study from a development project where the use of quantitative usability requirements was found useful. We used methods that do not exactly follow the existing well-known usability methods. We believe that this is not a unique case: most industrial development projects have specific constraints and

[5] Reasons not within the scope of this article.

limitations, and an ideal use of usability methods is not generally feasible. While we strongly recommend the use of measurable usability requirements, we do not propose our methods as a general solution. Clearly, each project has its specific features, and the usability methods should be selected and tailored based on the specific context of the project.

References

1. Norman DA, Draper S (eds) (1986) User centered system design. Hillsdale, Erlbaum (NY)
2. Gould JD, Boies SJ, Levy S, Richards JT, Schoonard J (1987) The 1984 Olympic message system: a test of behavioral principles of system design. Commun ACM 30(9):758–769
3. Beyer H, Holtzblatt K (1998) Contextual design: defining customer-centered systems. Morgan Kaufmann Publishers, San Francisco, p 472
4. Carroll JM, Mack RL (1985) Metaphor, computing systems and active learning. Int J Man Mach Stud 221(1):39–57
5. Nielsen J (1993) Usability engineering. Academic, San Diego, p 358
6. Hix D, Hartson HR (1993) Developing user interfaces: ensuring usability through product and process.Wiley, New York, p 416
7. Mayhew DJ (1999) The usability engineering lifecycle. Morgan Kaufman, San Fancisco
8. Constantine LL, Lockwood LAD (1999) Software for use. Addison-Wesley, New York, p 579
9. Cooper A, Saffo P (1999) The inmates are running the asylum: why high tech products drive us crazy and how to restore the sanity. Sams 261
10. Rosson MB, Carroll JM (2002) Usability engineering. Scenario-based development of human–computer interaction. Morgan Kaufmann Publishers, San Francisco
11. ISO/IEC, 13407 Human-centred design processes for interactive systems ISO/IEC 13407: 1999 (E)
12. Jokela T (2004) The KESSU usability design process model. Version 2.1. Oulu University, p 22
13. Good M, Spine TM, Whiteside J, G.P (1986) User-derived impact analysis as a tool for usability engineering. In: Conference proceedings on human factors in computing systems
14. Wixon D, Wilson C (1997) The usability engineering framework for product design and evaluation. In: Helander M, Landauer T, Prabhu P (eds) Handbook of human–computer interaction. Elsevier, Amsterdam. pp 653–688
15. Jokela T, Pirkola J (1999) Using quantitative usability goals in the design of a user interface for cellular phones. In: INTERACT '99 (Volume II). British Computer Society, Wiltshire, Edinborough
16. Göransson B, Gulliksen J, Boivie I (2003) The usability design process — integrating user-centred systems design in the software development process. Softw Process Improvement Practice 8(2)
17. Gulliksen J, Göransson B, Boivie I, Blomqvist S, Persson J, Cajander Å (2005) Key principles of user-centred systems design. In: Desmarais M, Gulliksen J, Seffah A (eds) Human-centered software engineering: bridging HCl, usability and software engineering
18. Wixon D (2003) Evaluating usability methods. Why the current literature fails the practitioner. Interactions 10(4):28–34
19. ISO/IEC (1998) 9241-11 Ergonomic requirements for office work with visual display terminals (VDT)s—Part 11 Guidance on usability. ISO/IEC 9241-11 :1998 (E)
20. NIST (2004) Proposed industry format for usability requirements. Draft version 0.62
21. Whiteside J, Bennett J, Holtzblatt K (1988) Usability engineering: our experience and evolution. In: Helander M (eds) Handbook of human–computer interaction. North-Holland, Amsterdam, pp 791–817
22. Kirakowski J, Corbett M (1993) SUMI: The software usability measurement inventory. Br J Educ Technol 24(3):210–212
23. Brooke J (1986) SUS — A "quick and dirty" usability scale. Digital Equipment Co. Ltd
24. Chin JP, Diehl VA, Norman KL (1988) Development of an instrument measuring user satisfaction of the human–computer interface. In: Proceedings of SIGCHI '88. New York
25. Jokela T (2005) Guiding designers to the world of usability: determining usability requirements through teamwork. In: Seffah A, Gulliksen J, Desmarais M (eds) Human–centered software engineering. Kluwer HCI series
26. Gould JD, Lewis C (1985) Designing for usability: key principles and what designers think. Commun ACM 28(3):300–311
27. Dumas JS, Redish JC (1993)A practical guide to usability testing. Ablex Publishing Corporation, Norwood
28. Bevan N, Macleod M (1994) Usability measurement in context. Behav Inf Technol 13(1,2):132–145
29. Macleod M, Bowden R, Bevan N, Curson I (1997) The MUSiC performance measurement method. Behav Inf Technol 16(4,5):279–293
30. Maguire M (1998) RESPECT user-centred requirements handbook. Version 3.3. HUSAT Research Institute (now the Ergonomics and Saftety Research Institute, ESRI), Loughborough University
31. Bevan N, Claridge N, Athousaki M, Maguire M, Catarci T, Matarazzo G, Raiss G (2002) Guide to specifying and evaluating usability as part of a contract, version1.0. PRUE project. Serco Usability Services, London, p 47
32. ANSI (2001) Common industry format for usability test reports. NCITS 354–2001
33. Butler KA (1985) Connecting theory and practice: a case study of achieving usability goals. In: SIGCHI 1985. ACM Press, New York, San Francisco
34. Karat J (1997) User-centered software evaluation methodologies, In: Helander MG, Landauer TK, Prabhu PV (eds) Handbook of human–computer interaction. Elsevier, Amsterdam
35. Card SK, Moran TP, Newell A (1983) The psychology of human–computer interaction. Lawrence Erlbaum Associates, Hillsdale
36. John BE (1995) Why GOMS?. In: Interactions pp 80–89
37. Preece J, Rogers Y, Sharp H (2002) Interaction design. Beyond human–computer interaction. Wiley, New York
38. Snyder C (2003) Paper prototyping. The fast and easy way to design and refine user interfaces. Morgan Kaufmann, San Francisco
39. Nielsen J (1994) Heuristic evaluation. In: Nielsen J, Mack RL (eds) Usability inspection methods. Wiley, New York
40. Thomas C, Bevan N (1996) Usability context analysis: a practical guide. Version 4.04. National Physical Laboratory, Teddington