



KUNGLTEKNISKA HÖGSKOLAN

Royal Institute of Technology
Numerical Analysis and Computer Science

CID-101, KTH, Stockholm, Sweden 2000

Bringing in the Social Perspective: User Centred Design

Jan Gulliksen

Jan Gulliksen

Bringing in the social perspective: User centred design

Report number: CID-101

ISSN number: ISSN 1403-0721 (print), 1403-073X (Web/PDF)

Publication date: August 2000

E-mail of author: Jan.Gulliksen@hci.uu.se

URL of author: <http://cid.nada.kth.se>

Reports can be ordered from:

CID, Centre for User Oriented IT Design

Nada, Dept. Computing Science

KTH, Royal Institute of Technology

S-100 44 Stockholm, Sweden

telephone: + 46 8 790 91 00

fax: + 46 8 790 90 99

e-mail: cid@nada.kth.se

URL: <http://www.nada.kth.se/cid/>

BRINGING IN THE SOCIAL PERSPECTIVE: USER CENTRED DESIGN

Jan Gulliksen

Department for Human Computer Interaction, Uppsala university,
Lägerhyddsvägen 18, SE-75237 Uppsala, Sweden. and Center for user-oriented
IT-design (CID), Royal Institute of Technology, Stockholm, Sweden.

1. Introduction

The social setting of end users does not only influence their task performance and attitudes towards computer systems but also their willingness to contribute their knowledge actively to the design of artefacts. Standish Group [Dennis, 1995] analysed IT-development projects in US showing that 31,1 % of the projects were aborted without having accomplished their goals, 52,7 % were completed with serious delays and at much higher costs (in general the costs increased with 189 %), and 16,2 % were completed within the budget and time frame. The most important factors for success in the projects that were completed within the timeframe were; user participation, management support, a clear requirement specification and good planning.

Successful software development requires *usable* systems. ISO 9241 defines Usability as; *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* [ISO 9241].

To design systems that are usable in a specific use situation, active involvement of representatives of the user population is essential, but no guarantee. Therefore user centred design is a necessity for designing usable systems. Our view on *user centred design (UCD)* of usable systems is based on ISO draft international standard on human centred design process for interactive systems [ISO 13407] and Gould & Lewis principles for designing for usability [Gould & Lewis, 1985], that is:

- *Work controlled development*. Early focus on users and tasks. The designer must understand the users, their cognitive behaviour, attitudes and the

characteristics of the work. Appropriate allocation of function between the user and the system is also important to prevent unnecessary control and preemption.

- *Active user participation* throughout the project, in analysis, design, development and evaluation. This requires a careful user selection process emphasising the skills of typical users, both:
 - Work domain experts (continuously through the development project)
 - Actual end-users (for evaluation of various design results)
- *Early prototyping* to evaluate and develop design solutions.
- *Continuous iteration* of design solutions. A cyclic process of design, evaluation and redesign should be repeated as often as possible. The evaluation process should mean empirical measurement in which experiments are performed with prototypes with which real users perform real tasks with the purpose of observing, monitoring and analysing the users' reactions and attitudes.
- *Multidisciplinary design teams*. Include a usability designer in the process.
- *Integrated design*. Continuous developments of the system, the work activity, help, education, organisation, etc. in the development work.

2. Common Problems with UCD

So, how come so many projects fail to reach their goals within the defined limit of time and budget? Several problems have been observed when having user participation in a development project [Gulliksen & Lantz, 1998] and we have classified them as:

- *Attitude problem*. Many system developers regard computer system development as an artistic occupation with an expressive task, or a task of breaking new technical limitations, rather than, as preferable, service in a work context.
- *Communication problems*. Trying to understand and interpret the worlds of the various roles involved in the development work, indicate a problem of communication. Power relations, group processes, communication languages, lacking time, ability and interest influence.
- *Methods and tools problems*. Although accessible to the public, methods and tools do not always support UCD and the design process.
- *Lacking time*. Iterations, as one of the fundamentals of UCD often tend to delay the project in the construction phase. Conversely, problems of

different types are generated by the fact that many large system development projects tend to last for several years.

- *Organisational problems.* Management seldom supports the allocation of sufficient time and resources for UCD. Adverse managerial influence, various conflicting power relations and the lack of minimal organisational support for usability-related work can present severe obstacles to this process.
- *Participants support.* There must be support for UCD both at the managerial level and at the user level. Time and resources need to be allocated to a project for the users to participate as much as necessary.
- *Competence problems.* The participants seldom have the knowledge, skill, special abilities or even interest in UCD. HCI knowledge is still difficult to apply in practical system development.
- *External aspects* Various unexpected incidents can disturb UCD work, e.g. processes of change, political or strategically important decisions. Differing interests can be represented, and conflicts may occur that requires external control.

Why do we face all of these different problems related to user participation and the usability problems with the resulting systems?

3. Overcoming obstacles to UCD

From several studies of UCD in practice we can observe difficulties in adopting UCD throughout the entire lifecycle. Usability related activities are usually adopted late in the process, evaluating the product right before it reaches the market, mostly without possibilities of correcting the problems.

3.1 Iterations and lacking development time

The difficulty adopting an iterative approach is an obstacle to UCD. One iteration must involve 1) a proper analysis of the user task and context, 2) a prototype design phase, and 3) a documented usability evaluation of the design prototype that produce evaluation results that need to be addressed in the following process. Merely claiming that usability aspects are considered, without performing documented evaluations with users, can not be regarded as iterative design.

When development projects are delayed, usability evaluations late in the process are very easy to exclude. Project management can therefore vote against iterative UCD for the fright of loosing control over the development times. Therefore, UCD must be adopted early in the process to be beneficiary and to prevent increasing development times later on.

3.2 Guidelines for User Participation and Selection

Despite extensive participation we see a lack of real communication and understanding between computer professionals and user representatives. The user population can be known or unknown, available or not, and different approaches need to be adopted depending on the user population. The important thing, however, is that the users that participate are skilled, dedicated and willing to contribute to changes. Depending on the user population various ways of selecting and using the user population in the development projects can be adopted.

By providing guidelines on user participation we address several of the problems related to communication, skills and attitudes in a UCD project. These domain specific guidelines describe whom, where, when, how and why users should participate, thereby providing practical knowledge useful for project management and development work. The guidelines give practical advice on the processes in which the users are involved. Like for example that it is important to address all aspects from the users and communicate the decisions made based on the user aspects back to the users to keep the users confidence.

3.3 Communication problems

How do we avoid having a HCI expert standing in one corner screaming? We propose the "Usability Designer" to promote UCD in system development projects. A usability designer should have extensive knowledge on human cognitive characteristics, abilities for aesthetic design, possibilities of understanding the work domain, some development tool knowledge and finally the social competence required to be a communication link between users and developers. Tool knowledge is essential to receive confidence with the developers. Understanding the work domain can only be achieved through participation in the preceding work modelling sessions. So far, in practice, this role has been shouldered by GUI-programmers, which unfortunately has led to several of the above mentioned problems.

3.4 Prototyping for a More Efficient Development Process

Typically in in-house development projects, up to a year of analysis work precedes the actual user interface design process. We pursue early prototyping as a means for efficient user participation and promote better iterative design of usable systems. So far, according to our observations, early user interface prototyping can make the user representatives feel contributing and make several of the other modelling steps faster and more efficient.

3.5 Holistic integrated design

Changing the technology evidently cause changes for the work activities, the organisational structures and the human beings with their skills and expertise [Leavitt, 1958]. However, this observation has not lead to any efforts to meet these challenges when developing the technology. We are currently striving for integrated design by defining methods for organisational change and learning simultaneously when developing computer support for a work situation. This requires an interdisciplinary research setting.

4. Conclusions

Successfully adopted UCD has, when adopted, been regarded as an absolute necessity to arrive at usable systems. However, few projects really adopt UCD throughout the entire development process. Traditional in-house development projects have a well defined user population, participating as a part of their work. With the recent increase in network services for the public, the user population has become all the more difficult to specify and access. We need to develop new and efficient ways of capturing user requirements, performing user centred prototyping and usability evaluations for these new technologies. As researchers we here have the responsibility to educate software developers on UCD and enhance the possibilities for efficient user participation to be able to increase the maturity and awareness of these questions.

5. References

Standish Group, (1995) *The Chaos Report*. Dennis MA, USA, Available: <http://www.standishgroup.com/chaos.html>

International Organisation of Standardisation (1998) ISO IS 9241 *Ergonomic requirements for office work with visual display terminals (VDTs) Part 11 – Guidance on usability*. (International standard)

International Organisation of Standardisation (1998) ISO/DIS 13407 *Human centred design process of interactive systems*. (Draft international standard)

Gould, J.D., & Lewis, C. (1985). Designing for Usability: Key Principles and What Designers Think. *Comm. of the ACM*, Vol. 38, No. 3, pp. 300-311.

Gulliksen, J. & Lantz, A. (1998) User orientation – does it work or is it just an aggravation? *I³-magazine* No. 2, March 1998, The European Network for Intelligent Information Interfaces.

Leavitt, H.J. (1958) *Managerial Psychology*. University of Chicago Press, Ltd. London.