Int. J. Human-Computer Studies (2001) **55,** 587–634
doi:10.1006/ijhc.2001.0503
Available online at http://www.idealibrary.com on IDEAL®

Methods to support human-centred design

MARTIN MAGUIRE
HUSAT Research Institute, Loughborough University, Leicestershire LE11 3TU, UK.
email: m.c.maguire@lboro.ac.uk

(Received 9 January 2001, and accepted in revised form 26 June 2001)

This paper notes the importance of usable systems and promotes the process of human-centred design as a way to achieve them. Adopting the framework of ISO 13407, each of the main processes in the human-centred design cycle is considered in turn along with a set of usability methods to support it. These methods are briefly described with references to further information. Each set of methods is also presented in a table format to enable the reader to compare and select them for different design situations.

KEYWORDS: human-centred design; user-centred design; usability; design methods; ISO 13407.

1. The importance of usable systems

Usability is now widely recognized as critical to the success of an interactive system or product (Shackel, 1981, 1984; Eason, 1984; Whiteside, Bennett & Holtzblatt, 1988; Fowler, 1991; Shackel, 1991; Nielsen, 1994; ISO, 1997*b*). Many poorly designed and unusable systems exist which users find difficult to learn and complicated to operate. These systems are likely to be under used, misused or fall into disuse with frustrated users maintaining their current working methods. The outcome is costly for the organization using the system, and harmful to the reputation of the company which developed and supplied it.

The benefits of designing a usable system can be summed up as follows.

- **Increased productivity.** A system designed following usability principles, and tailored to the user's preferred way of working, will allow them to operate effectively rather than lose time struggling with a complex set of functions and an unhelpful user interface. A usable system will allow the user to concentrate on the *task* rather than the *tool*.
- **Reduced errors**. A significant proportion of "human error" can often be attributed to a poorly designed user interface. Avoiding inconsistencies, ambiguities or other interface design faults will reduce user error.
- **Reduced training and support**. A well-designed and usable system can reinforce learning, thus reducing training time and the need for human support.
- **Improved acceptance**. Improved user acceptance is often an indirect outcome from the design of a usable system. Most users would rather use, and would be more likely to

trust, a well-designed system which provides information that can be easily accessed and presented in a format which is easy to assimilate and use.
- **Enhanced reputation**. A well-designed system will promote a positive user and customer response, and enhance the developing company's reputation in the marketplace.

This paper discusses how usable systems can be achieved via a human-centred approach to design and presents a range of usability methods that can be employed to support this process. In the past, these methods, although tested and well established, were often used separately by different practitioners and in isolation. However, there are now frameworks for integrating the techniques and aligning them with the software development process (ISO 14598), i.e. ISO 13407 (ISO, 1999) and ISO TR 18529 (ISO, 2000*a*). This paper takes the ISO 13407 human-centred design framework as a basis for showing how different methods can be used together to support a human-centred design process.

2. Human-centred design

Within the field of software development, there are numerous methods for designing software applications. All stress the need to meet technical and functional requirements for the software. It is equally important to consider the user requirements if the benefits outlined above are to be realized. Human-centred design (HCD) is concerned with incorporating the user's perspective into the software development process in order to achieve a usable system.

2.1. KEY PRINCIPLES OF HCD

The HCD approach is a complement to software development methods rather than a replacement for them. The key principles of HCD are as follows.

- *The active involvement of users and clear understanding of user and task requirements.* One of the key strengths of human-centred design is the active involvement of end-users who have knowledge of the context in which the system will be used. Involving end-users can also enhance the acceptance of and commitment to the new software, as staff come to feel that the system is being designed in consultation with them rather than being imposed on them. Strategies for achieving user involvement are discussed by Damodaran (1996).
- *An appropriate allocation of function between user and system.* It is important to determine which aspects of a job or task should be handled by people and which can be handled by software and hardware. This division of labour should be based on an appreciation of human capabilities, their limitations and a thorough grasp of the particular demands of the task.
- *Iteration of design solutions.* Iterative software design entails receiving feedback from end-users following their use of early design solutions. These may range from simple paper mock-ups of screen layouts to software prototypes with greater fidelity. The users attempt to accomplish "real world" tasks using the prototype. The feedback from this exercise is used to develop the design further.

- *Multi-disciplinary design teams*. Human-centred system development is a collaborative process which benefits from the active involvement of various parties, each of whom have insights and expertise to share. It is therefore important that the development team be made up of experts with technical skills and those with a "stake" in the proposed software. The team might thus include managers, usability specialists, end-users, software engineers, graphic designers, interaction designers, training and support staff and task experts.

2.2. HCD DEVELOPMENT CYCLE

According to the ISO 13407 standard on human-centred design (ISO, 1999) there are five essential processes which should be undertaken in order to incorporate usability requirements into the software development process. These are as follows.

- Plan the human-centred design process.
- Understand and specify the context of use.
- Specify the user and organizational requirements.
- Produce designs and prototypes.
- Carry out user-based assessment.

The processes are carried out in an iterative fashion as depicted in Figure 1, with the cycle being repeated until the particular usability objectives have been attained.

Sections 3–7 describe each process in turn and present a set of methods or activities that can support them; these are listed in Table 1. The compilation of this list of methods

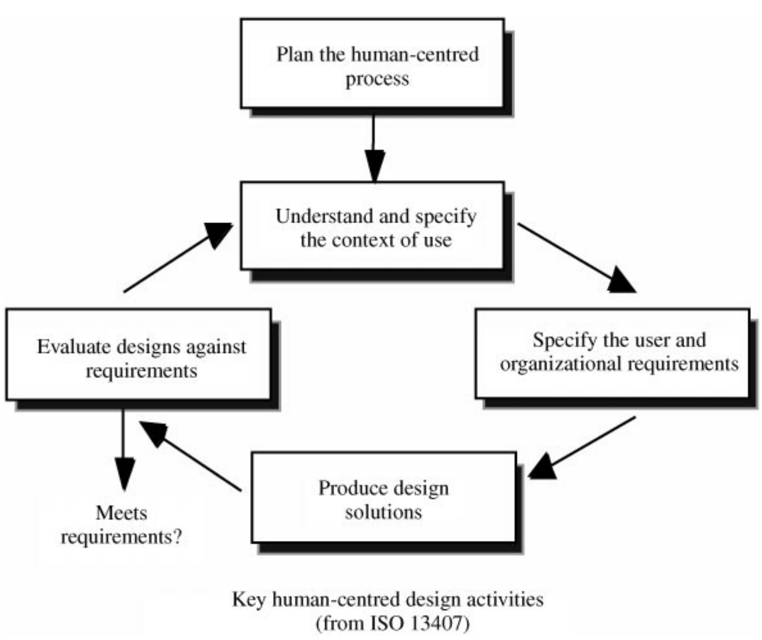

FIGURE 1. The human-centred design cycle.

TABLE 1
Methods for human-centred design

Planning (Section 3)	Context of use (Section 4)	Requirements (Section 5)	Design (Section 6)	Evaluation (Section 7)
3.1. Usability planning and scoping	4.1. Identify stakeholders	5.1. Stakeholder analysis	6.1. Brainstorming	7.1. Participatory evaluation
3.2. Usability cost–benefit analysis	4.2. Context of use analysis	5.2. User cost–benefit analysis	6.2. Parallel design	7.2. Assisted evaluation
	4.3. Survey of existing users	5.3. User requirements interview	6.3. Design guidelines and standards	7.3. Heuristic or expert evaluation
	4.4. Field study/user observation	5.4. Focus groups	6.4. Storyboarding	7.4. Controlled user testing
	4.5. Diary keeping	5.5. Scenarios of use	6.5. Affinity diagram	7.5. Satisfaction questionnaires
	4.6. Task analysis	5.6. Personas	6.6. Card sorting	7.6. Assessing cognitive workload
		5.7. Existing system/competitor analysis	6.7. Paper prototyping	7.7. Critical incidents
		5.8. Task/function mapping	6.8. Software prototyping	7.8. Post-experience interviews
		5.9. Allocation of function	6.9. Wizard-of-Oz prototyping	
		5.10. User, usability and organizational requirements	6.10. Organizational prototyping	

draws upon the experience of the HUSAT Research Institute and the EC UsabilityNet project (Bevan, Kirakowski, Claridge, Granlund & Strasser, 2001). In each of Sections 3–7 the methods are described briefly with references for further information. Tables are also provided to compare the main characteristics of each method to assist the reader with their selection. Section 8 discusses the important additional process of system release and management of change, while Section 9 provides examples of the application of human-centred design methods to system design projects.

3. Planning the human-centred design process

If the application of a human-centred design approach is to be successful, it must be carefully planned and managed throughout all parts of the system development process. For example, involving usability expertise in some specific parts but not others will be inadequate. HCD plays an important role in a project by reducing the risk of system failure by maintaining the effective flow of information about users to all relevant parts of a project team. It is crucial to ensure full integration of the HCD activities as part of the system strategy for the whole of the project (cf. Booher, 1990; Damodaran, 1998; MoD, 2000; ISO, 2000*a*; Earthy, Sherwood Jones & Bevan, 2001).

The first step is to bring together the stakeholders to discuss and agree how usability can contribute to the project objectives and to prioritize usability work (Section 3.1). It may then be necessary to perform a study to establish the potential benefit to be gained from including HCD activities within the system development process and which methods to use and can be afforded (Section 3.2). Both of these activities are described next.

3.1. USABILITY PLANNING AND SCOPING

This strategic activity is best performed by bringing together in a meeting all the stakeholders relevant to the development, to create a common vision for how usability can support the project objectives. It links the business goals to the usability goals, ensures that all factors that relate to use of the system are identified before design work starts, and identifies the priorities for usability work.

The objective is to collect and agree high-level information about the following.

- Why is the system being developed? What are the overall objectives? How will it be judged as a success?
- Who are the intended users and what are their tasks? Why will they use the system? What is their experience and expertise? What other stakeholders will be impacted by the system?
- What are the technical and environmental constraints? What types of hardware will be used in what environments?
- What key functionality is needed to support the user needs?
- How will the system be used? What is the overall workflow (e.g. from instructor preparation, through student interaction, to instructor marking)? What are the typical scenarios of how and why users will interact with the system?

- What are the usability goals? (How important is ease of use and ease of learning? How long should it take users to complete their tasks? Is it important to minimize user errors? What GUI (graphical-user interface) style guide should be used?)
- How will users obtain assistance in using the system?
- Are there any initial design concepts?

This will identify the areas that need to be explored in more depth later. One output of the meeting is a usability plan that specifies the structures to support the usability work, i.e.

- Those responsible for performing usability activities (ideally a multidisciplinary team).
- Those who will represent the users, the involvement they will have in the design process, and any training they require.
- The lines of communication in performing usability work between usability specialists, users, customers, managers, developers, marketing, etc. This will include how to get information about the project and product to those responsible for usability, in an agreed format.

3.2. USABILITY COST–BENEFIT ANALYSIS

The aim of this activity is to establish the potential benefits of adopting a human-centred approach within the system design process. The cost–benefits can be calculated by comparing the costs of user-centred design activities with the potential savings that will be made during development, sales, use and support.

The extent of the financial benefits will depend on how completely human-centred design can be implemented. A balance needs to be obtained so that a convincing case can be made for benefits that are substantially larger than the costs of additional user-centred activities. Vendors can benefit in development, sales and support, purchasers can benefit in use and support, and systems developed for in-house use can benefit in development, use and support. Development savings can be made as a result of the following.

- Reduced development time and cost to produce a product which has only relevant functionality and needs less late changes to meet user needs.
- Reduced cost of future redesign of the architecture to make future versions of the product more usable.

Sales revenue may increase as a result of an increased competitive edge, repeat purchases from more satisfied customers and higher ratings for usability in the trade press.

Usage savings may be made as a result of reduced task time and increased productivity, reduced user errors that have to be corrected later, reduced user errors leading to an improved quality of service, reduced training time for users and reduced staff turnover as a result of higher satisfaction and motivation. Support savings may be made as a result of reduced costs of producing training materials, reduced time providing training, reduced time spent by other staff providing assistance when users encounter difficulties and reduced help line support.

Another way to look at the issue is to balance the cost of the allocation of resources to HCD against the benefit of lowered risk of system and/or project failure.

TABLE 2
Methods for planning the human-centred design process

Method	Summary, aims and benefits	When to apply	Typical (min.) time	Approach of method
3.1. Usability planning and scoping	Links usability to project objectives and prioritizes usability work	At the very start. Strategic activity to initiate usability work	4 days (2 days)	Meeting with key stakeholders
3.2. Usability cost–benefit analysis (Bias & Mayhew, 1994)	Establishes the potential benefits of adopting a human-centred approach and the targets for usability work	Useful to help cost justify usability work at the start of a project	3 days (2 days)	Meeting held with project manager, usability specialist and user representative(s)

For organizations already committed to human-centred design, a cost–benefit analysis is not essential but it can provide a valuable input when formulating a usability plan. The technique can be used repeatedly as a development project progresses to reassess the importance of various activities. The process can also be used to compare different usability methods and so aid selection of the most cost-effective method.

More information can be found in Bias and Mayhew (1994) and Bevan (2001).

3.3. SUMMARY OF METHODS FOR PLANNING AND FEASIBILITY

Table 2 summarizes the methods for use in planning a human-centred design process. For each method the following information is given.

- Summary description with an indication of aims and benefits.
- When the method is best used.
- Estimate of typical and minimum time required to perform it (including preparation time e.g. to set up meetings, recruit subjects, etc.).
- General approach of the method.

Estimates of the typical (and minimum) effort in person days required by usability specialists to apply each method are given. For a large project some activities may need to be repeated for different parts of the project. To achieve minimum effort requires someone with sufficient expertise to tailor the essential aspects of the method to local needs.

4. Understand and specify the context of use

When a system or product is developed, it will be used within a certain context. It will be used by a user population with certain characteristics. They will have certain goals and wish to perform certain tasks. The system will also be used within a certain range of technical, physical and social or organizational conditions that may affect its use.

The quality of use of a system, including usability and user health and safety, depends on having a very good understanding of the context of use of the system. For example, the design of a bank machine (or ATM) will be much more usable if it is designed for use at night, in bright sunlight and by people in wheelchairs. Similarly, in an office environment there are many characteristics which can impinge on the usability of a new software product (e.g. user workload, support available or interruptions). Capturing contextual information is therefore important for helping to specify user requirements as well as providing a sound basis for later evaluation activities.

For well-understood systems, it may be sufficient to identify the stakeholders and arrange a meeting to review the context of use. For more complex systems this may need to be complemented by a task analysis and a study of existing users.

4.1. IDENTIFY STAKEHOLDERS

It is important to identify all the users and other stakeholders who may be impacted by the system. This will help to ensure that the needs of all those involved are taken into account and, if required, the system is tested by them. User groups may include end-users, supervisors, installers and maintainers, other stakeholders (those who influence or are affected by the system) including recipients of output from the system, marketing staff, purchasers and support staff (see Taylor, 1990).

4.2. CONTEXT-OF-USE ANALYSIS

This is a structured method for eliciting detailed information about the context of use for a system as a foundation for later usability activities, particularly user requirements specification and evaluation. Stakeholders attend a facilitated meeting, called a Context Meeting, to help complete a detailed questionnaire. The information collected provides details of the characteristics of the users, their tasks and their operating environment. The main elements of a context analysis are shown in Table 3.

This is a simple technique to use when most of the information is already known by the stakeholders. To avoid prolonging the meeting, when using such a detailed checklist, it is important to complete in advance any items that are not contentious and highlight the issues that need to be discussed.

For the simplest systems, the context information can be collected as part of the stakeholder identification meeting, using a less structured process. If it is impossible to arrange a meeting, the information can be gathered by interviewing the stakeholders or using a questionnaire. This has the disadvantage that there is no opportunity to establish consensus on, and commitment to, the usage characteristics. In more complex situations, where the information is not well understood, field studies and contextual design may be required to collect and analyse the information.

Context-of-use analysis was one of the outcomes of the ESPRIT HUFIT project, Human Factors in Information Technology (Allison, Catterall, Dowd, Galer, Maguire & Taylor 1992). A set of tools was developed for identifying user types, their needs and characteristics, and translating this information into user requirements (Taylor, 1990). In the ESPRIT MUSiC project this was developed further and the "Usability Context Questionnaire" (Maissel, Dillon, Maguire, Rengger & Sweeney, 1991) was created. This

TABLE 3
Context-of-use factors

User group	Tasks	Technical environment	Physical environment	Organizational environment
• System skills and experience • Task knowledge • Training • Qualifications • Language skills • Age and gender • Physical and cognitive capabilities • Attitudes and motivations	• Task list • Goal • Output • Steps • Frequency • Importance • Duration • Dependencies	• Hardware • Software • Network • Reference materials • Other equipment	• Auditory environment • Thermal environment • Visual environment • Vibration • Space and furniture • User posture • Health hazards • Protective clothing and equipment	• Work practices • Assistance • Interruptions • Management and communications structure • Computer use policy • Organizational aims • Industrial relations • Job characteristics

is a paper-based questionnaire to assist in the capture of context-of-use information and the specification of the conditions for an evaluation. A guidebook for context analysis was later developed by Thomas and Bevan (1995). See also the paper on context-of-use analysis in this journal issue (Maguire, 2001*a*).

4.3. SURVEY OF EXISTING USERS

A survey involves administering a set of written questions to a sample population of users. Surveys can help determine the needs of users, current work practices and attitudes to new system ideas. Surveys are normally composed of a mix of "closed" questions with fixed responses and "open" questions, where the respondents are free to answer as they wish. This method is useful for obtaining quantitative (as well as some qualitative) data from a large number of users about existing tasks or the current system. For further information see Preece, Rogers, Sharp, Benyon, Holland and Carey (1994).

4.4. FIELD STUDY/OBSERVATION

Observational methods involve an investigator viewing users as they work and taking notes of the activity which takes place. Observation may be either direct, where the investigator is actually present during the task, or indirect, where the task is recorded on video-tape by the analysis team and viewed at a later time. The observer tries to be unobtrusive during the session and only poses questions if clarification is needed. Obtaining the cooperation of users is vital, so the interpersonal skills of the observer are important. For further information see Preece *et al*. (1994).

TABLE 4
Comparison of context-of-use methods

Method	Summary and benefits	When to apply	Typical (min.) time	Approach to the method
4.1. Identify stakeholders Taylor (1990)	Lists all users and stakeholders for the system. Ensures that no one is omitted during system design	Should be applied for all systems. For generic systems, it may be supplemented with a market analysis of customers	0.5 day (0.5 day)	Meeting held with project manager and user representatives
4.2. Context-of-use analysis (Thomas & Bevan, 1995; Maguire, 2001*a*,*b*)	Provides background (context) information against which design and evaluation takes place	Needed for all systems	2 days (1 day)	Meeting with representatives from each main user group, and design team representative
4.3. Survey of existing users (Preece *et al*., 1994)	Questionnaire distributed to a sample population of future users. Provides quantitative data from large number of users	When there is diverse user population When users are difficult to access because of location, role or status When quantitative data needed, e.g. functional preferences	15 days (6 days)	Develop survey explaining aims at the start. Pilot questions and instructions carefully. Provide mix of open and closed questions. Keep as short as possible. Provide return envelope

4.4. Field study/user observation (Preece *et al.*, 1994)	Investigator views users as they work and takes notes on the activity taking place. Provides data on current system usage and context for system	When situation is difficult for user to describe in interview or discussion. When environmental context has significant effect on usability	8 days (5 days)	Establish objectives and type of events to be recorded. Establish ground rules, timescales and that observer is to be ignored. Be unobtrusive and give those observed time off from study
4.5. Diary keeping (Poulson *et al.*, 1996)	To record user behaviour over a period of time to gain a picture of how future system can support the user	When there is a current system or when it is necessary to obtain data about current user activity	15 days (8 days)	Create and test the diary format. Distribute to users to complete at certain times or during certain activities
4.6. Task analysis (Kirwan & Ainsworth, 1992)	The study of what a user is required to do in terms of actions and/or cognitive processes to achieve a task	When it is important to understand task actions in detail as a basis for system development	15 days (6 days)	Start with map of users and list their main roles. Identify individual users. Plan meetings and possibly observation sessions. Check understanding with users

4.5. DIARY KEEPING

Activity diaries provide a record of user behaviour over a period of time. They require the participant to record activities they are engaged in throughout a normal day. The information may lead to the identification of user requirements for a new system or product. Diaries may contain both structured multiple choice questions and open-ended sections, where the participant can record events in their own words. Diaries may be recorded on paper, on video tape, or on-line via input forms administered by computer. For further information see: Poulson, Ashby and Richardson (1996).

4.6. TASK ANALYSIS

Task analysis can be defined as the study of what a user is required to do in terms of actions and/or cognitive processes to achieve a task. A detailed task analysis can be conducted to understand the current system and the information flows within it. Understanding these information flows and user actions is important if appropriate system features and functions are to be developed. Failure to allocate sufficient resources to task analysis increases the potential for costly problems arising in later phases of development. Task analysis makes it possible to design and allocate tasks appropriately within the new system (see Section 5.9). The functions to be included within the system and the user interface can then be appropriately specified.

There are many variations of task analysis and notations for recording task activities. One of the most widely used is hierarchical task analysis, where high-level tasks are de-composed into more detailed components and sequences (Shepherd, 1985, 1989). Another approach is to create a flow chart showing the sequence of human activities and the associated inputs and outputs (Ericsson, 2001). Kirwan and Ainsworth (1992) have produced a guide to the different task analysis methods, and Hackos and Redish (1998) explain some of the simpler methods for user interface design.

4.7. COMPARISON OF METHODS FOR SPECIFYING THE CONTEXT OF USE

Table 4 provides a comparison of context-of-use methods. This may be used to help select the appropriate methods for different system design situations.

5. Specify the user and organizational requirements

Requirements elicitation and analysis is widely accepted to be the most crucial part of software development. Indeed, the success of a software development programme can largely depend on how well this activity is carried out. A recent survey performed by the Standish Group (http://standishgroup.com/visitor/chaos.htm) in the United States showed that the two most common causes of system failure were: insufficient effort to establish user requirements and lack of user involvement in the design process. It has also been found that for e-commerce websites, user success in purchasing ranges from as little as 25–42%, much of this relating to an inability to find the required product to buy in a reasonable time. It was also recently estimated by Phil Terry, CEO of Creative Good, that in the year 2000 there were $19 billion of lost sales in the United States due to usability problems of e-commerce sites.

These problems highlight a failure to recognize the needs of the system user and to specify them in a way that designers can incorporate within the system development process.

General guidance on specifying user and organizational requirements and objectives is provided in ISO 13407 (ISO, 1999). This states that the following elements should be covered in the specification.

- Identification of the range of relevant users and other personnel in the design.
- Provision of a clear statement of design goals.
- An indication of appropriate priorities for the different requirements.
- Provision of measurable benchmarks against which the emerging design can be tested.
- Evidence of acceptance of the requirements by the stakeholders or their representatives.
- Acknowledgement of any statutory or legislative requirements, for example, for health and safety.
- Clear documentation of the requirements and related information. Also, it is important to manage changing requirements as the system develops.

The following sections describe general methods that can be used to support user and organizational requirements specification.

5.1. STAKEHOLDER ANALYSIS

Stakeholder analysis identifies, for each user and stakeholder group, their *main roles, responsibilities and task goals* in relation to the system. For example, a public information system situated in a local library or advice bureau must be designed to meet the contrasting needs of the general public, the information service staff and information providers. The general public will have the goal to retrieve information by browsing or to answer a specific query (so will require an intuitive simple interface enabling the system to be used on a "walk-up and use" basis). Information service staff will have the goal of monitoring system usage, performing simple maintenance tasks and providing adequate support for the general public users. Information providers have the goal of inputting information into the system in a convenient manner. Stakeholder analysis is described in Damodaran, Simpson and Wilson (1980).

5.2. USER COST–BENEFIT ANALYSIS

User cost–benefit analysis is a method for comparing the costs and benefits for different user groups when considering a new system to serve different user groups. The proposed roles of each user group are considered and the costs and benefits under specific headings are listed and quantified. This provides an overview of how acceptable each user group will find the new system. It also provides the opportunity to rethink the system design or user roles to provide a more acceptable solution for all groups. A process for performing a user cost–benefit analysis is described by Eason (1988).

5.3. USER REQUIREMENTS INTERVIEWS

Interviewing is a commonly used technique where users, stakeholders and domain experts are asked questions by an interviewer in order to gain information about their needs or requirements in relation to the new system. Interviews are usually semi-structured based on a series of fixed questions with scope for the user to expand on their responses. Semi-structured interviewing is useful in situations where broad issues may be understood, but the range of respondents' reactions to these issues is not fully known. Structured interviewing should only be carried out in situations where the respondents' range of replies is already well known and there is a need to gauge the strength of each shade of opinion. Interviews can also be used as part of a task analysis (Section 4.6). For further information see Preece *et al.* (1994) and Macaulay (1996).

5.4. FOCUS GROUPS

A focus group brings together a cross-section of stakeholders in a discussion group format. This method is useful for requirements elicitation and can help to identify issues which need to be tackled. The general idea is that each participant can act to stimulate ideas in the other people present, and that, by a process of discussion, the collective view becomes established which is greater than the individual parts. Focus groups are not generally appropriate for evaluation (Nielsen, 2000*a*). For further information see: Caplan (1990), Blomberg, Giacomi, Mosher and Swenton-Hall (1993), Preece *et al.* (1994), Macaulay (1996), Poulson *et al.* (1996), Farley (1997) and Bruseberg and McDonagh-Philp (2001).

5.5. SCENARIOS OF USE

Scenarios give detailed realistic examples of how users may carry out their tasks in a specified context with the future system. The primary aim of scenario building is to provide examples of future use as an aid to understanding and clarifying user requirements and to provide a basis for later usability testing.

Scenarios encourage designers to consider the characteristics of the intended users, their tasks and their environment, and enable usability issues to be explored at a very early stage in the design process (before a commitment to code has been made). They can help identify usability targets and likely task completion times. The method promotes developer buy-in and encourages a human-centred design approach.

Scenarios should be based on the most important tasks from the context-of-use information. They are best developed in conjunction with users. User goals are decomposed into the operations needed to achieve them. Task time estimates and completion criteria can be added to provide usability goals. For further information see Clark (1991), Nielsen (1991) and van Schaik (1999).

5.6. PERSONAS

Personas are a means of representing users' needs to the design team, by creating caricatures to represent the most important user groups. Each persona is given a name, personality and picture. They are particularly valuable when it is difficult to include user representatives in the design team. Each persona can be associated with one or more scenarios of use. Potential design solutions can then be evaluated against the needs of a particular persona and the tasks in the scenarios. Personas are popular with innovative

Functionality matrix						
Name of system	Functions within system					
Users and tasks	F1	F2	F3	F4	F5	Comments
User A						
Task A	●	●				
Task B			○			
Task C					○	
User B						
Task A		●			○	
Task B			●		○	
Task C				○		
Function selection	High priority	High priority	High priority	Low priority	Medium priority	

FIGURE 2. Structure for functionality matrix. ● = Critical to task; ○ = Occasional use.

design groups, where they are used to stimulate creativity rather than refine a design solution. The use of personas is described in Cooper (1999).

5.7. EXISTING SYSTEM/COMPETITOR ANALYSIS

Evaluating an existing or competitor version of the system can provide valuable information about the extent to which current systems meet user needs and can identify potential usability problems to avoid in the new system. Useful features identified in a competitor system can also be fed into the design process (Section 6). Measures of effectiveness, efficiency and satisfaction can be used as a baseline for the new system. To obtain accurate measures a controlled user test (Section 7.4) should be used, but valuable information can still be obtained from less formal methods of testing.

5.8. TASK/FUNCTION MAPPING

This process specifies the system functions that each user will require for the different tasks that they perform. The most critical task functions are identified so that more time can be spent on them during usability testing later in the design process. Figure 2 shows how this can be done with a Functionality Matrix (Catterall, 1990). It is important that input from different user groups is obtained in order to complete the matrix fully. This method is useful for systems where the number of possible functions is high (e.g. in a generic software package) and where the range of tasks that the user will perform is well specified. In these situations, the functionality matrix can be used to trade-off different functions, or to add and remove functions depending on their value for supporting specific tasks. It is also useful for multi-user systems to ensure that the tasks of each user type are supported.

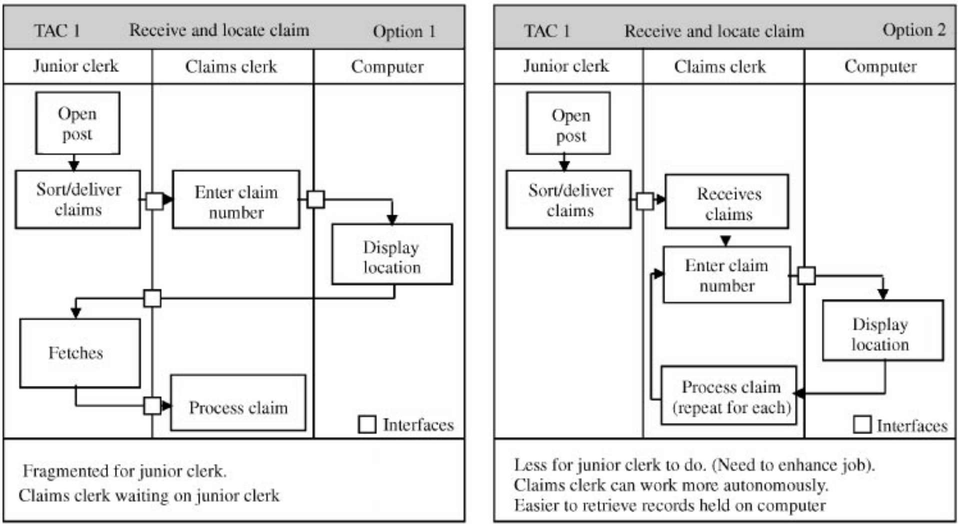

FIGURE 3. Allocation of function chart.

5.9. ALLOCATION OF FUNCTION

A successful system depends on the effective "allocation of function" between the system and the users—as ISO (1999) 13407 states in Section 7.3.2, allocation of function is "the division of system tasks into those performed by humans and those performed by technology". Different task allocation options may need to be considered before specifying a clear system boundary. A range of options are established to identify the optimal division of labour, to provide job satisfaction and efficient operation of the whole work process. The use of task allocation charts is most useful for systems which affect whole work processes rather than single user, single task products. Figure 3, taken from Ip, Damodaran, Olphert and Maguire (1990), shows two allocation options for a process involving different levels of computer storage of records. In option 1, a Junior Clerk in a welfare benefits organization opens the post, delivers the claims to the Claims Clerk who enters each client's identification number into the computer. The computer then displays the location of the client records (i.e. filing cabinet numbers) which the Junior Clerk then fetches for the Claims Clerk to process. In option 2, the computer holds the records on file. The Junior clerk sorts out the claims, delivers them to the Claims Clerk who then calls them up on the computer as and when he or she wishes to process them.

5.10. USER, USABILITY AND ORGANIZATIONAL REQUIREMENTS

5.10.1. User requirements. It is important to establish and document the user requirements so that they lead into the process of designing the system itself. User requirements will include summary descriptions of the tasks that the system will support and the functions that will be provided to support them. It will provide example task scenarios and possible interaction steps for the future system, and describe any features of the system to meet the context-of-use characteristics.

5.10.2. Usability requirements. It is also necessary to describe the detailed usability requirements in order to set objectives for the design team, and help prioritize usability work. (Broad usability objectives may already have been established in the usability planning and scoping activity.) The general usability goals to define are the following.

- Effectiveness: the degree of success with which users achieve their task goals.
- Efficiency: the time it takes to complete tasks.
- Satisfaction: user comfort and acceptability.

These are most easily derived from the evaluation of an existing system, and are independent of any specific implementation. Other more detailed usability issues provide more specific design objectives.

- Understandability: whether users understand what the system can do.
- Learnability: the training, time and effort required to learn to use the system.
- Operability or supportiveness: supporting the users throughout the interaction and helping them to overcome problems that may occur.
- Flexibility: enabling tasks to be carried out in different ways to suit different situations.
- Attractiveness: encouraging user interest and motivating them to explore the system.

Having established usability requirements, it is then necessary to translate the requirements into a specification (specification = requirement + measure). For more information see ISO (2000*b*) which provides a framework for specifying measurable requirements.

5.10.3. Organizational requirements. A third element is to specify the organizational requirements for the user-system complex. Organizational requirements are those that come out of a system being placed into a social context (i.e. a set of social and organizational structures) rather than those that derive from the functions to be performed and the tasks to be assisted. An understanding of organizational requirements will help to create systems that can support the management structure of the organization and communications within it, as well as group and collaborative working. Defining and grouping the tasks in an appropriate way will help to create motivating and satisfying jobs, ideally allowing users autonomy, flexibility, provision of good feedback on their performance and the opportunity to develop their skills and careers. Organizational requirements can be derived from an understanding of current power structures, obligations and responsibilities, levels of control and autonomy, as well as values and ethics. Relevant statutory or legislative requirements, including those concerned with safety and health, may also be classed as organizational requirements.

The information needed to specify user, usability and organizational requirements will be drawn from the context of use and requirements activities described in Sections 4 and 5. As design proceeds, prototypes of the system will be developed which are then evaluated. The requirements can then be enhanced from the results with prototype versions of the system; this enables the requirements to be made more concrete, specific and more readily satisfied.

The EU RESPECT project (Maguire, 1998) has produced a handbook which offers a framework for capturing user requirements. This includes a step-by-step approach and methods for gathering user requirements data and templates for recording the data. To help understand organizational requirements, the EU ORDIT Project (Olphert & Harker, 1994) developed a framework within which user organizations and system developers can communicate about both organizational and technical issues—an important element in specifying organizational requirements. Another leading publication is by Robertson and Robertson (1999), who provide a comprehensive text on performing requirements analysis, and on a step-by-step approach called the Volere method. Within another article in this journal Roberston (2001) describes many innovative methods for "trawling for requirements" and gives the web address for downloading Volere.

5.11. COMPARISON OF METHODS FOR SPECIFYING USER AND ORGANIZATIONAL REQUIREMENTS

Table 5 provides a comparison of methods described to support the specification of user and organizational requirements. It can be used to help select the appropriate methods and combination of methods for different situations.

6. Produce design solutions

Design solutions arise in many ways—from copying and development, by logical progression from previous designs, through to innovative creativity. Whatever the original source, all design ideas as they progress will go through iterative development. Mock-ups and simulation of the system are necessary to support this iterative design lifecycle. At the simplest, they may consist of a series of user interface screens and a partial database allowing potential users to interact with, visualize and comment on the future design. Such simulations or prototypes can be produced both quickly and easily in the early stages of the system development cycle for evaluation by human factors experts, user representatives and members of the design team. Changes to the design may then be made rapidly in response to user feedback, so that major problems with the design can be identified before system development begins. This helps to avoid the costly process of correcting design faults in the later stages of the development cycle. Such a development setting is generically called Rapid Application Development (RAD).

The following list of methods includes techniques for generating ideas and new designs (brainstorming and parallel design), use of design guidelines and standards (to ensure compliance with legal requirements), and techniques for representing future systems (storyboarding, paper-based prototyping, computer-based prototyping, Wizard-of-Oz and organizational prototyping).

Although a range of prototyping methods are described, this is not intended to imply that all forms must be used for the development of every product. As a minimum, a low fidelity prototype should be developed (e.g. paper mock-up or scripted demonstration), followed by a high fidelity prototype (working simulation or operational system) (cf. Hall, 2001). This will help to ensure that the product will both be usable and will meet the functional needs of the users. The process of iterative prototyping requires that the features of the prototype, the way it addresses key requirements, and the nature of

TABLE 5
Comparison of requirement methods

Method	Summary	When to apply and benefits	Typical (min.) time	Approach to the method
5.1. Stakeholder analysis (Damodaran *et al.*, 1980)	Use list of all users and stakeholders for the system, from Section 4.1, to study and analyse their roles and responsibilities. Ensures that no one is omitted during system design	Should be applied for all systems. For generic systems, the analysis may be supplemented with a market analysis	1 day (0.5 day)	Obtain overview of stakeholder roles from project manager based on stakeholders identified in Section 4.1. Then organise interviews or meeting with stakeholders to discuss roles in more detail
5.2. User cost–benefit analysis (Eason, 1988)	Compares costs and benefits for each user group. Helps provide acceptable user roles and avoid de-skilling	Applicable mainly for bespoke systems when there are several user groups and stakeholders for the system	2 days (1 day)	Meeting with representatives from each main user group, and design team representatives
5.3. User requirements interviews (Macaulay, 1996)	Provides individual views on user requirements from a range of users. Face-to-face approach enables in depth questioning	Useful for all systems	8 days (5 days)	Decide on list of issues to be covered with each user. Allow user to give initial thoughts before moving to interview questions. Telephone interviews may be performed when users hard to access
5.4. Focus groups (Caplan, 1990)	To bring together group of stakeholders to discuss possible requirements	Generally useful for all systems	14 days (8 days)	A discussion agenda has to be developed and tested. Homogeneous but contrasting user groups are recruited to discuss requirements

TABLE 5
Continued

Method	Summary	When to apply and benefits	Typical (min.) time	Approach to the method
5.5. Scenarios of use (Nielsen, 1991)	Development of characterizations of users and their tasks in a specified context	Generally useful for all systems. Help users to understand the way the future system might work and to specify their requirements in concrete terms	6 days (3 days)	Develop example scenarios, possibly with images and pictures. Hold meeting with experienced facilitator and technical representative to clarify possibilities
5.6. Personas (Cooper, 1999)	Detailed caricatures used to represent user needs	To highlight users issues when users cannot participate in design	2 days (1 day)	Develop a detailed profile of the motivations and tasks of a typical representative of each key user group
5.7. Existing system/ competitor analysis	Evaluate an existing or competitor system to baseline usability	Valuable to highlight existing usability problems and set objectives	4 days (2 days)	Carry out a controlled or assisted evaluation
5.8. Task/function mapping (Catterall, 1990)	A process which specifies the system functions that each user will require for the different tasks	Helps to clarify which functions are needed. Useful for generic products where a wide range of functions could be included. Acts as a way to exclude less important functions	6 days (4 days)	User tasks and contextual characteristics entered down left-hand column of matrix. Possible functions are listed along the top. Matching process takes place at a design team meeting

TABLE 5
Continued

Method	Summary	When to apply and benefits	Typical (min.) time	Approach to the method
5.9. Allocation of function (Ip *et al.*, 1990)	Different task allocation options considered between users, stakeholders and the system, before specifying a clear system boundary	Helps establish system boundary for effective performance which also helps create acceptable and interesting human roles and jobs	10 days (5 days)	For each task or work flow, select those where alternative allocations are possible and draw chart for each. Discuss each option in a user/stakeholder meeting
5.10. User, usability and organizational requirements (ISO, 2000*b*)	Establishes main benefits for usability design work	Needed for all systems. Alongside hardware and software requirements, helps to set total goals for a good system	4 days (2 days)	Meeting with user representatives, marketing, and design team

the problems identified and changes made are properly documented. This will allow the iteration of design solutions to be managed effectively.

6.1. BRAINSTORMING

Brainstorming brings together a set of design and task experts to inspire each other in the creative, idea generation phase of the problem-solving process. It is used to generate new ideas by freeing the mind to accept any idea that is suggested, thus allowing freedom for creativity. The method has been widely used in design. The results of a brainstorming session are, it is hoped, a set of good ideas and a general feel for the solution area. Clustering methods may be used to enhance the outcome of a group session. Brainstorming is particularly useful in the development phase when little of the actual design is known and there is a need for new ideas. For more information, see Jones (1980) and Osborn (1963).

6.2. PARALLEL DESIGN

It is often helpful to develop possible system concepts with parallel design sessions in which different designers work out possible designs. Using this approach, several small groups of designers work independently, since the goal is to generate as much diversity as possible. The aim is to develop and evaluate different system ideas before settling on a single solution (possibly drawing from several solutions) as a basis for the system. For more information see Nielsen (1993).

6.3. DESIGN GUIDELINES AND STANDARDS

Designers and HCI specialists may refer to design guidelines for guidance on ergonomic issues associated with the system being developed. The ISO 9241 standard (ISO, 1997*a*) covers many aspects of hardware and software user-interface design, and contains the best and most widely agreed body of software ergonomics advice. The processes recommended in Part 1 and Annex 1 in each of parts 12–17 of the standard ensure a systematic evaluation of each clause to check its applicability to the particular system being developed. This approach complements the use of style guides which provide more specific guidance. There are also several papers providing user interface design guidelines for specific applications such as kiosk systems (Maguire, 1999*a*) and inclusive design for the disabled and elderly (Nicolle & Abascal, 2001).

Style guides embody good practice in interface design, and following a style guide will increase the consistency between screens and can reduce the development time. For a graphic user interface (GUI) careful research performed within companies has produced good and stable guidelines, so the operating system style guide should be followed whenever possible. For websites, design guidelines are still evolving rapidly and being tested on public sites. Eventually, the best designs will survive and the bad ones will decline as users abandon poorly designed sites (Nielsen, 2000*b*, p. 218). Websites should define their own style guide based on good web design principles (e.g. Nielsen, 2000*b*).

6.4. STORYBOARDING

Storyboards, also termed "Presentation Scenarios" by Nielsen (1991), are sequences of images which show the relationship between user actions or inputs and system (e.g. screen) outputs. A typical storyboard will contain a number of images depicting features such as menus, dialogue boxes and windows. The formation of these screen representations into a sequence conveys further information regarding the possible structures, functionality and navigation options available. The storyboard can be shown to colleagues in a design team as well as to potential users, allowing others to visualize the composition and scope of possible interfaces and offer critical feedback. Few technical resources are required to create a storyboard. Simple drawing tools (both computer and non computer-based) are sufficient. Storyboards also provide a platform for exploring user requirements options via a static representation of the future system by showing them to potential users and members of a design team. This can result in the selection and refinement of requirements. See Nielsen (1991), Madsen and Aiken (1993) and Preece *et al.* (1994) for more information.

6.5. AFFINITY DIAGRAM

Affinity diagramming is a simple technique for organizing the structure of a new system: designers or users write down potential screens or functions on sticky notes and then organize the notes by grouping them and by placing closely related concepts close to each other. It is especially useful for uncovering the structure and relationships in a poorly understood domain. Affinity diagrams are often a good next step after a brainstorming session. See Beyer and Holtzblatt (1998) for more information. Related "card sorting" techniques are useful for uncovering similar groupings from users.

6.6. CARD SORTING

Card sorting is a technique for uncovering the hierarchical structure in a set of concepts by having users group items written on a set of cards; this is often used, for instance, to work out the organization of a website. For a website, users would be given cards with the names of the web pages on the site and asked to group the cards into related categories. After doing so, the users may be asked to break down their groups into subgroups for large sites. After gathering the groupings from several users, designers can typically spot clear organizations across many users. Statistical analysis can uncover the best groupings from the data where it is not clear by inspection, though inconsistent groupings may be a sign of a poorly defined goal for the website or a poor choice of web page names. More information can be found in McDonald and Schvaneveldt (1988).

6.7. PAPER PROTOTYPING

Designers create a paper-based simulation of user interface elements (menus, buttons, icons, windows, dialogue sequences, etc.) using paper, card, acetate and pens. When the paper prototype has been prepared, a member of the design team sits before a user and "plays the computer" by moving interface elements around in response to the user's actions. The difficulties encountered by the user and their comments are recorded by an observer and/or on video or audio tape. More information is available from Nielsen (1991) and Rettig (1994).

One variant of paper-prototyping is to video-tape the testing of the paper interface as the elements are moved and changed by members of the design team. This is sometimes called *video prototyping*. End-users do not interact directly with the paper prototype but can later view the video representation. This approach can be useful for demonstrating interface layout and the dynamics of navigation—particularly to larger audiences. More information is available within Vertelney (1989) and Young and Greenlee (1992).

6.8. SOFTWARE PROTOTYPING

This approach to prototyping utilizes computer simulations to provide a more realistic mock-up of the system under development. Software prototypes provide a greater level of realism than is normally possible with simple paper mock-ups. Again, end-users interact with the prototype to accomplish set tasks and any problems which arise are noted. Many software development packages now include a visualization or screen development facility to allow software prototypes to be created quickly. This can be used to establish an acceptable design for the user but is then thrown away prior to full implementation.

Most web site development packages also have a direct screen creation facility which supports a prototyping approach. However, when the software is implemented, it is desirable to allow flexibility within the design process to allow for further change. Some design processes are based on a rapid application development (RAD) approach. Here a small group of designers and users work intensively on a prototype, making frequent changes in response to user comment. The prototype rapidly evolves towards a stable solution which can then be implemented. For larger systems, the small groups may work in this way on different components of the system which are then integrated. This requires a clearly defined total structure, distinct functional boundaries, and an agreed interface style. For more details, see Wilson and Rosenberg (1988) and Preece *et al.* (1994).

The different prototyping representations described in Sections 6.4 and 6.7 to 6.8 are appropriate for different stages in the design process. The potential role of different prototypes is summarized in Figure 4, based on Maguire (1996).

6.9. WIZARD-OF-OZ PROTOTYPING

This method is a variant of computer-based prototyping. A user interacts with a computer system that is actually operated by a hidden developer—referred to as the "wizard". The wizard processes inputs from the user and responds with simulated system output. The approach is particularly suited to exploring design possibilities which are demanding to implement such as intelligent interfaces possibly featuring agents or advisors, and/or natural language processing. See also Gould, Conti and Hovanyecz (1983), Maulsby, Greenberg and Mander (1993), and Nielsen (1993).

6.10. ORGANIZATIONAL PROTOTYPING

An organizational prototype can be created by simulating a working environment (e.g. a job centre or benefit office) (cf. Olphert & Damodaran, 1991) and prototyping the

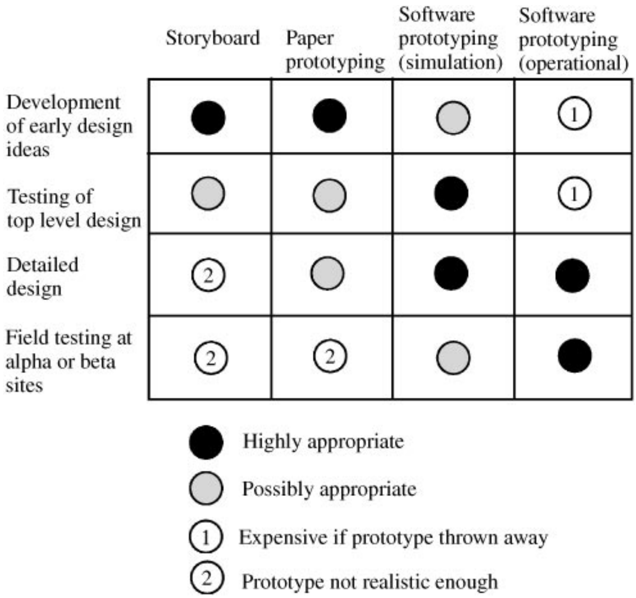

FIGURE 4. Application of different prototyping methods.

communication and information flows between users in different roles. The roles may be fulfilled by future end-users, design team members and future customers (e.g. members of the public) following predefined scenarios. Prototype computer systems may also be used although this may not be necessary in the early stages of development (cf. Eason & Olphert, 1995), and mock-ups of computer technology may be sufficient. This approach will help to determine how well the system development supports human activities, and whether an appropriate allocation of function has been defined.

Gale and Christie (1987) developed the concept of the Controlled Adaptive Experimental Flexible Office of the Future in an ecological Environment (CAFÉ OF EVE). Their idea was to establish a true working environment but as a laboratory with video cameras, on-line questionnaires, and the capture of performance data via the keyboard, etc. Workers would be employed to carry out their jobs and at the end of the day reflect on and give feedback on the systems they use. The strength of the CAFÉ OF EVE is that it would combine the rigour of laboratory work with the ecological validity of the field environment.

6.11. COMPARISON OF METHODS TO SUPPORT DESIGN SOLUTION DEVELOPMENT

Table 6 provides a comparison of design and prototyping methods. This may be used to help select the appropriate methods for different system design situations.

7. Evaluate designs against requirements

Designs should be evaluated throughout development, initially using low fidelity (typically paper) prototypes, followed later by more sophisticated prototypes. This is a very

TABLE 6
Comparison of methods to assist in producing design solutions

Method	Summary	When to apply and benefits	Typical (min.) time	Approach to the method
6.1. Brainstorming (Jones, 1980)	Brings together a set of experts to inspire each other in the creative design phase of system design	Early in design process. Useful when the design area is very open and there is an opportunity to develop an innovative system	3 days (2 days)	Assemble group of people with a variety of expertise. At the meeting, keep ideas short, and record all ideas rather than criticize them. Encourage a large number of ideas
6.2. Parallel design (Nielsen, 1993)	Several small groups work on the same problem to produce designs in parallel	Early in design process. Useful way to generate several concrete designs in a short time	6 days (3 days)	Set up 2 or more design groups with roughly equal number in each. Provide clear list of requirements and specify means of output e.g. structure as a series of PowerPoint™ screens
6.3. Design guidelines and standards	Designers and HCI specialists review usability design standards and style guides to feed into design process	Should be applied soon after a design concept is developed, before detailed design commences. Makes design team familiar with good practice	5 days (2 days)	Design concept is documented and design issues listed. Each member of workshop may be asked to review one standard before the meeting. At the meeting, design issues are discussed in relation to standards and guides
6.4. Storyboarding (Nielsen, 1991)	Sequences of images are created which demonstrate the relationship between user inputs and system outputs (e.g. screen outputs)	Allows users to visualize and comment upon future user interface designs and the functions provided	6 days (4 days)	Consider scenarios to illustrate with storyboards. Represent them with high-level screen sequences to avoid distraction. Discuss in design team/user meeting

6.5. Affinity diagram (Beyer & Holtzblatt, 1998)	Use sticky notes to organize screens or functions from a user perspective	Early in the design process to organize the interface from a user perspective	3 days (2 days)	Use a group of developers and users to discuss and agree on the best structure
6.6. Card sorting (McDonald & Schvaneveldt, 1988)	Sort items written on cards into a hierarchical structure	Early in the design process to group data from a user perspective	3 days (2 days)	Users arrange cards into groupings that can be analysed to find common patterns
6.7. Paper prototyping (and video prototyping) (Rettig, 1994)	Designers create a paper-based simulation of an interface to test interaction with a user. One variant is to video the paper prototype interactions and show to users to comment	Quick way to create a prototype and perform "user test". A PowerPoint™ version may also be developed as an alternative to paper	4 days (2 days)	Create prototype with simple materials. Test with person controlling prototype in response to requested actions from the user
6.8. Software prototyping (Preece *et al.*, 1994)	Computer simulations are developed to represent system under development in a realistic way	Gives users a more realistic experience of the "look and feel" of the future design	12 days (for development and small scale testing) (8 days)	Develop software structure and screen layouts on paper before developing the software prototype. Select appropriate users to test prototype in controlled or less formal setting—see section 7 on evaluation methods
6.9. Wizard-of-Oz prototyping (Maulsby *et al.*, 1993)	Involves a user interacting with a computer system that is actually operated by a hidden developer	Suitable to explore design possibilities that are difficult to implement such as expert systems and natural language interaction. Allows designer acting as wizard to gain user insights	12 days (for development and small-scale testing) (8 days)	Prepare system, realistic task scenarios, system responses and recording facilities. Test to ensure wizard can react appropriately. Make clear to user testers that system is being operated by another person
6.10. Organizational prototyping (Eason & Olphert, 1995)	A simulation of processes in the user environment to explore how user actions integrate with the new computer system	Applicable for bespoke systems where operational procedures need to be tested. Helps to define acceptable user roles	8 days (5 days)	Plan the room layout and organizational processes to be prototyped. Provide clear scripts for each person involved. Ensure that observers can record events unintrusively. Activities videoed to review later

important activity within the system development lifecycle; it can confirm how far user and organizational objectives have been met as well as provide further information for refining the design. As with the other human-centred activities it is advisable to carry out evaluations from the earliest opportunity, before making changes becomes prohibitively expensive.

There are two main reasons for usability evaluation.

- To improve the product as part of the development process (by identifying and fixing usability problems): "formative testing".
- To find out whether people can use the product successfully: "summative testing".

Problems can be identified by any of the methods in this section. User-based methods are more likely to reveal genuine problems, but expert-based methods can highlight shortcomings that may not be revealed by a limited number of users. User-based testing is required to find out whether people can use a product successfully.

When running user tests, the emphasis may be on identifying system problems and feeding them quickly into the design process (formative testing). A small number of test sessions may be sufficient for this purpose with the evaluator observing system users and making notes. There may be some prompting and assistance if the user gets stuck. The technique can be used to identify the most significant user-interface problems, but it is not intended to provide reliable metrics.

Alternatively, the main aim may be to derive metric data (for summative testing). Here, the real-world working environment and the product under development are simulated as closely as possible. Users undertake realistic tasks while observers make notes; timings are taken; and video and/or audio recordings are made. Generally, the observer tries to avoid interacting with the user apart from guiding the test session. The observations are subsequently analysed to derive metrics. Design problems are also identified.

There are essentially three levels of formality when performing evaluation studies: *participative* (least formal), *assisted* (intermediate) and *controlled evaluation* (most formal).

It is important to identify and fix usability problems early in the design process and less formal methods are most cost effective. When it is important to understand how the user is thinking, a *participatory* approach is appropriate and questioning may include asking the user for their impressions of a set of screen designs, what they think different elements may do, and what they expect the result of their next action to be. The user may also be asked to suggest how individual elements could be improved.

An *assisted* approach may be adopted where the user is requested to perform tasks and is invited to talk aloud. However, the evaluator only intervenes if the user gets stuck. The objective is to obtain the maximum feedback from the user while trying to maintain as realistic an operational environment as possible.

To find out how successful users will be with the full working system a *controlled* user test is required, as closely as possible replicating the real world in the test environment, only making available any assistance that the user might actually have (e.g. possibly a manual or a help line). The controlled user test can be used to evaluate whether usability requirements have been achieved, for example via the following measures.

• Effectiveness: the degree of success with which users achieve their task goals.
• Efficiency: the time it takes to complete tasks.
• Satisfaction: user comfort and acceptability.

User-based testing can take place in a controlled laboratory environment, or at the user's work place. The aim is to gather information about the user's performance with the system, their comments as they operate it, their post-test reactions and the evaluator's observations.

Many IT organizations such as IBM, Hewlett-Packard, DEC and Fujitsu (ICL) have invested in advanced, dedicated laboratories for performing such usability evaluation work. This facility may consist of a user area which can be set up to-reflect a range of operational contexts and a control area for observation by the human factors evaluator. A one-way mirror may separate the two areas so that the user can be observed by the evaluator in the control area although the evaluator cannot be seen by the user (Sazegari, Rohn, Uyeda, Neugebaurer & Spielmann, 1994).

While usability evaluations require care in their planning and performance, in practice, they often need to be carried out within a short timescale as part of an iterative development cycle. Here prototypes are often provided by a commercial client organization and changes are made to it as a basis for further user testing. Early analysis of usability studies (Nielsen & Landauer, 1993) showed that 85% of major usability problems could be identified after testing with 5–8 users. However, recent work (Spool & Schroeder, 2001) has shown that this does not apply to e-commerce web sites where the complex range of different tasks require larger numbers of users to identify major problems. In their study, serious problems were still being found on the 13th and 15th user test sessions. Also, where there are considerable numbers of potential user types, there needs to be sufficient numbers of each type in the evaluation plan.

A more formal evaluation will typically involve running controlled user test sessions with at least eight users. User interactions and comments can be recorded during each test session on videotape for later analysis. The output of a usability evaluation is normally a report describing the process of testing that was carried out, the results obtained, and recommendations for system improvement. An additional and useful technique is to create a short film, 5–15 min long, composed of video clips from the user sessions to illustrate key problems that were encountered with the prototype or facilities that work especially well. This provides a means of emphasizing the results of the evaluation to the design team. The results may also be passed on to other departments such as marketing or senior management to support the case for the development of a new product or innovative set of features.

At present, a Common Industry Format (CIF) for reporting usability test results is currently being agreed between major software suppliers and purchasers in the United Sates in an initiative coordinated by NIST (the US National Institute of Standards and Technology). The aim of this initiative is to make usability a more concrete issue for consumers and suppliers and to provide a means of reporting the results of a usability evaluation in a standard way (Bevan, 1999). It is also intended to provide confidence that a developing project meets a specified set of usability criteria and enhances the communication between the customer of a system and the supplier on usability issues. HUSAT is a partner within the PRUE project (Providing Reports on Usability Evaluation), a take-up action supported by the EC (IST-1999-20692), which is trialling the CIF format with European industry (PRUE, 2001).

A range of evaluation methods is described in the following sections. These start from the more exploratory formative methods, employed during the early stages of prototype development, continuing to the more formal summative testing as the prototype develops through usability work.

7.1. PARTICIPATORY EVALUATION

Users employ a prototype as they work through task scenarios. They explain what they are doing by talking or "thinking-aloud". This information is recorded on tape and/or captured by an observer. The observer also prompts users when they are quiet and actively questions them with respect to their intentions and expectations. For more information see Monk, Wright, Haber and Davenport (1993).

An *evaluation workshop* is a variant of participatory evaluation. Users and developers meet together and the user representatives try to use the system to accomplish set tasks while designers observe. The designers can later explore the issues identified through a facilitated discussion. Several trials can be run to focus on different system features or versions of the system. The method is applicable to a wide range of computer applications and especially custom developments with known stakeholders. One strength of the technique is that it brings users and developers together in a facilitated environment. Multi-user involvement will draw out several perspectives on a particular design issue. For more information on one version of this approach, see Fitter *et al.* (1991).

Another form of participatory evaluation is an *evaluation walkthrough*. This is a process of going step-by-step through a system design, getting reactions from relevant staff, typically users. A human factors specialist should facilitate the walkthrough although a member of the design team may operate it, while one or more users will comment as the walkthrough proceeds. A list of problems is drawn up by consensus and corresponding severity ratings are defined as they arise. When the design elements have been examined, the problem list and severity levels should be reviewed and changes should be proposed. For more information, see Maulsby *et al.* (1993), Nielsen (1993).

7.2. ASSISTED EVALUATION

An assisted evaluation is one where the user is invited to perform a series of tasks and is observed by a human factors specialist who records users' problems and comments, and events of interest. The user is asked to try and complete the tasks without help, although the evaluator may give prompts if the user gets stuck. This form of evaluation allows the evaluator to assess how well the system supports the user in completing tasks but also provides the option for the user to provide some feedback as they proceed. If appropriate, video-tape recording for subsequent analysis may be used.

7.3. HEURISTIC OR EXPERT EVALUATION

Heuristic or expert evaluation is a technique where one or more usability and task experts will review a system prototype and identify potential problems that users may face when using it. There are of course dangers in employing just one expert as he/she could be affected by personal biases, and experience has shown that one expert may not capture all the major problems. Therefore, it is recommended that multiple experts be employed.

The experts will start by gaining a thorough understanding of the user characteristics, the nature of the task and the working environment, in discussion with the design team and desirably also with user representatives. The expert will then study the prototype or demonstration of the system and mentally pose a number of questions which will highlight problems and lead to recommendations for improving it. At the same time, the expert may identify problems instinctively, i.e. where some feature contrasts with the expert's view of good practice. They may evaluate the system with reference to established guidelines or principles, noting down their observations and often ranking them in order of severity.

The main advantage of an expert appraisal is that it is a quick and easy way to obtain feedback and recommendations. The disadvantages are that experts may have personal biases towards specific design features, and it is often hard to set aside one's expertise and assume the role of the user. Authors such as Nielsen and Shneiderman offer checklists of design rules or heuristics to help prompt the evaluator and provide a structure for reporting design problems. For more information, see Nielsen (1992), Nielsen and Landauer (1993) and Shneiderman (1998).

7.4. CONTROLLED USER TESTING

The most revealing method of usability evaluation is to set up system trials where representative users are asked to perform a series of tasks with it. This may be set up in a controlled laboratory environment, at the developer's site or in the field within a customer organization. The aim is to gather information about the users' performance with the system, their comments as they operate it, their post-test reactions and the evaluator's observations. A controlled user testing study to evaluate a prototype will typically involve running test sessions with 8–25 users.

The main benefit of this approach is that the system will be tested under conditions close to those that will exist when it is used "for real". While technical designers and human factors experts may diagnose a large proportion of potential system problems, experience has shown that working with users will reveal new insights that can affect the system design.

Data from user trials can be captured in a number of ways.

- Automatic system monitoring may be set up whereby the system itself records interaction events of importance. These can be time-stamped to provide accurate information about the user's performance or their methods of navigating through the system.
- An evaluator observes and manually records events during the interaction session. These may include: time to complete task, points of apparent user difficulty, user comments, errors made, number of reversals through the interface, number of times assistance is required, demeanour of the user, approach to using the system, etc. While this method is very demanding, it means that useful data are recorded immediately on paper from which results can be obtained straightaway.

- A third method is to record each user session onto videotape. This technique has proved very useful since a comprehensive record of a session can be made and then analysed at leisure to gather both user performance data, behaviour and verbal comments during the test session.

The ESPRIT 5429 MUSiC project (Measuring Usability in Context) developed a set of standard tools and techniques for measuring software "quality of use" or usability (Bevan & Macleod, 1994). These tools incorporated a set of clearly defined methods and metrics for investigating different aspects of usability. One of the main outcomes from MUSiC was the performance measurement method (PMM). This included a usability context analysis (UCA) questionnaire and a structured method to evaluate user performance with the test system. This was achieved by direct observation and by capturing the user sessions on videotape. Video analysis and support software (called DRUM) was developed to help calculate measurements across the user sample. The performance metrics included user effectiveness, efficiency, relative user efficiency (compared to an expert) and productive period (productive time not spent in overcoming problems and seeking help).

For more general information on user testing see Nielsen (1993), Dumas and Redish (1993), Lindgaard (1994) and Maguire (1996).

7.5. SATISFACTION QUESTIONNAIRES

User subjective questionnaires capture the subjective impressions formed by users, based on their experiences with a deployed system or new prototype. This can be achieved with the use of questionnaires or through direct communication with the respondents. Following the use of a system, people fill in a standardized questionnaire and their answers are analysed statistically. Examples of questionnaires include SUMI (Kirakowski, 1996), WAMMI (Kirakowski & Claridge, 2001), QUIS (Chin, Diehl & Norman, 1988), SUS (Brooke, 1996) and SAQ (Maguire, 2001*b*). For SUMI, as well as a global assessment of usability, the questionnaire data provides information on: perceived efficiency, affect (likeability), control, learnability, helpfulness; and an assessment of how these results compare with results for similar software (deduced from a database of past results).

7.6. ASSESSING COGNITIVE WORKLOAD

Measuring cognitive workload involves assessing how much mental effort a user expends whilst using a prototype or deployed system. For example, this can be obtained from questionnaires such as the Subjective Mental Effort Questionnaire (SMEQ) and the Task Load Index (TLX). The SMEQ has one scale which measures the amount of effort people feel they have invested in a given task. The TLX has six scales (mental, physical, temporal, performance, effort and frustration) to measure the individual's perception of what a task has asked of them. It is also possible to collect objective data from heart rate variability and respiration rate. For more information contact: WITlab—Work and Interaction Technology Laboratory, Delft University of Technology, Jaffalaan 5, 2628 RZ Delft, The Netherlands.

7.7. CRITICAL INCIDENTS

Critical incidents are events that represent significant failures of a design. Verbal reports of the incident are analysed and categorized to determine the frequency of different incident categories. This enables design deficiencies to be identified. It can highlight the importance of improving features supporting a very infrequent but important task that otherwise might get ignored by other methods. It can be a very economical way of gathering data, but relies on the accuracy of users' recall. Automatic system monitoring may be set up whereby the system itself records interaction events of importance. These can be time-stamped to provide accurate information about the users' performance or their methods of navigating through the system. For further information see Flanagan (1954), Galdo, Williges and Williges (1986) and Carroll, Koenemann-Belliveau, Rosson and Singley (1993).

7.8. POST-EXPERIENCE INTERVIEWS

Individual interviews are a quick and inexpensive way to obtain subjective feedback from users based on their practical experience of a system or product. The interviews may be based on the current system they are using or be part of a debriefing session following testing of a new prototype. The interviewer should base his/her questions on a pre-specified list of items while allowing the user freedom to express additional views that they feel are important. For further information see Preece *et al.* (1994) and Macaulay (1996).

7.9. COMPARISON OF METHODS TO EVALUATE DESIGN SOLUTIONS

Table 7 provides a comparison of the methods described to evaluate designs against user and organisational requirements. This may be used to help select the appropriate methods for different situations.

8. System release and management of change

Effort devoted to careful user analysis, usability design and testing can be wasted by poor delivery to end-users. If a good product is delivered with poor support, or if it is used in ways that were not envisaged in design (through inadequate user requirements specification), it can be rendered unusable. Installation and user support for a new product or system can be seen to relate to both its supplier and its customer. The techniques and processes that each should follow can be summarized as follows (based on Maguire & Vaughan, 1997):

Processes for the system **supplier** are as follows.

- Assisting the installation process and training the user to maintain the system.
- Providing effective technical support following implementation.
- Provision of documentation and on-line help.
- Setting up of User Groups to support the user in initial and continued use of the product.

TABLE 7

Comparison of methods to evaluate designs against user requirements

Method	Summary	When to apply and benefits	Typical (min.) time	Approach to the method
7.1. Participatory evaluation (Monk *et al*., 1993)	Here the user works through the system, possibly performing tasks or exploring it freely. These are prompted and assisted by the evaluator as required	Provides a means to identify user problems and misunderstandings about the system	8 days (4 days)	Task scenarios for the user to follow are prepared. For each user session, one evaluator guides the session and prompts the user while another records observations and user comments
7.1a. Evaluation workshop (Fitter *et al*., 1991)	A participatory form of evaluation where users and developers meet together. User representatives try to use the system to accomplish set tasks	An intense session of user testing which can produce results quickly. It brings users and developers together in a facilitated environment	6 days (3 days)	A series of user sessions are run, possibly focusing on different aspects of the system. Users and designers watch the sessions and discuss the results
7.1b. Evaluation walkthrough or discussion (Nielsen, 1993)	A walkthrough is a process of going step-by-step through a system design and getting reactions from user representatives	Useful when detailed feedback is required at a detailed level	6 days (3 days)	Decide on the issues or task scenarios to be covered in the walkthrough. Assign roles of facilitator, system controller and notetaker before holding the meeting
7.2. Assisted evaluation	The user is invited to perform a series of tasks and is observed by a human factors specialist who records users problems, events of interest and user comments	Provides an idea of how well users can operate a system with minimal help while also giving some verbal feedback	9 days (5 days)	Define a suitable range of users and typical set of tasks. For each user session evaluator observes the user and only helps if necessary. User comments are recorded and analysed and may be prompted

7.3. Heuristic or expert evaluation (Nielsen, 1992)	One or more usability and task experts will review a system prototype and identify potential problems that users may face when interacting with it	As a first step to identify the major problems with a system before user testing. The approach can also be applied to an existing system as a basis for developing a new system	3 days (2 days)	One or more experts will review the system either separately or together. They should perform a context-of-use analysis before performing the evaluation. Usability checklists or style guides can support the process
7.4. Controlled user testing (Dumas & Redish, 1993; Bevan & Macleod, 1994)	Users test the prototype system in controlled conditions, performing representative tasks and providing verbal feedback. Performance measures may be taken	Shows how the system prototype will operate when exposed to "real use". Allows collection of usability performance measures	16 days (10 days)	Careful planning is required to recruit representative users, create realistic task scenarios and define a user session procedure. Piloting is required to ensure procedure and recording mechanisms work smoothly
7.5. Satisfaction questionnaires	Questionnaires capture the subjective impressions formed by users, based on their experiences with a system or new prototype	Quick and inexpensive way to measure user satisfaction	4 days (2 days)	One person will administer the questionnaire and between 8 and 20 users will usually act as respondents. They should be representative of the end-users to obtain valid results

Table 7
Continued

Method	Summary	When to apply and benefits	Typical (min.) time	Approach to the method
7.6. Assessing cognitive workload	Assessment of the level of mental effort a user expends whilst using a prototype or deployed system. Uses a questionnaire or physiological measures	Useful in environments when system user is under stress	8 days (4 days)	Observe environment beforehand and determine appropriate points to measure workload. Ensure that evaluator does not intrude and add to workload
7.7. Critical incidents (Galdo *et al*., 1986; Carroll *et al*., 1993)	Critical events that result in errors and user problems are recorded	Highlights system features that may cause errors and problems	10 days (6 days)	Users interviewed to recount incidents. User reports cross checked to validate data gathered
7.8. Post-experience interviews (Preece *et al*., 1994)	Users provide feedback on the current system they are using or after system testing	Quick and inexpensive way to obtain subjective feedback from users	4 days (3 days)	Decide on list of issues to be covered in interview. Allow user to give initial thoughts before moving to interview questions

Factors related to the system **customer management** are as follows.

- Making users aware of the forthcoming system and its impact on their work.
- User involvement.
- Provision of training to support initial and continued learning for users.
- Provision of health, safety and workplace design activities.
- The working environment.
- Carrying out user audits to capture feedback on the product in use.
- Managing organizational change. (This last factor is described in more detail below.)

Modern telecommunications products often only benefit people if they are accompanied by changes in the way people work, e.g. teleworking and mobile communications. To sustain a product's usability and acceptability, the supplier may need to support the processes of organizational change and system configuration as the product is installed.

Some products are much more likely to be associated with major organizational changes than others. Where changes will be required to get the benefits of products, user organizations may need help in understanding the need for change and in making the changes.

The support may need to cover five areas.

- The user organization may need to set new organizational objectives and set out policies, practices and time scales for their implementation, for example, teleworking.
- Organizational changes may need to be made at the time of product implementation, for example, decisions may need to be taken about who has responsibility for sustaining databases and ensuring back-up.
- Physical workstation and environment issues may need to be addressed in order to comply with Display Screen Equipment Regulations (DSE, 1992).
- The product may need to be configured for use, and facilities allocated to users according to the requirements of their job.
- In addition to providing the right facilities it may be necessary to support group working via networks to specify "read rights" and "write rights" to protect confidentiality, information ownership and security. Plans should be developed to cover these issues.

A comprehensive guide on organizational change has been produced by Eason (1988). HUSAT was also a partner on the ESPRIT AIT IMPLANT project which developed a change management methodology called GENCMM (Generic ENterprise Change Management Methodology) for large organizations (see Vaughan *et al*., 1988; and Allabrune *et al*., 1999).

9. Examples of the application of usability methods

This section describes examples of the application of usability methods within a number of different IT projects that involved the author.

9.1. EVALUATION OF HOME PRODUCTS

9.1.1. Purpose. The HUSAT Research Institute participated in the European project FACE "Familiarity Achieved through Common user interface Elements" (Burmester, 1997). The aim was to produce general design rules for consumer products. It involved collaboration with major companies manufacturing consumer electronics. The approach adopted was to simulate a range of alternative designs for different home product systems. These included a video recorder, a heating controller and a security system.

9.1.2. Selection and employment of methods. The process involved performing a **user survey (4.3)** to understand the attitudes of consumers towards existing devices and to gauge the importance of different usability characteristics. Since the range of users is potentially very large, a survey of 400 consumers was performed with samples of 100 in the UK, Germany, France and Italy to obtain a broad European perspective. **Interviews (5.3)** were then performed to help study the way they use devices at present and to understand the problems they face with their own products at home. These interviews were carried out on a sample of 40 users who owned a variety of consumer products from different manufacturers. This provided a broad perspective on what facilities people liked and disliked and which were easier to use or more difficult to use.

From this work, a series of design rules for simpler usage were defined. These were implemented within a set of alternative software prototypes of new interface designs. The companies produced initial versions for Human Factors experts to comment on. This feedback was incorporated into revised simulations which were then tested with members of the public using **assisted evaluations (7.2)** performing tasks such as programming a video recorder or setting up a heating control programme. User attitudes were also measured using **satisfaction questionnaires (7.5)**. The results enabled comparisons of prototype designs to be made and the best features of each design to be identified. These features fed into the refinement of the user interface design rules.

9.1.3. Constraints. Practical problems that needed to be addressed were mainly due to time constraints on test sessions. Users were asked to look at a device simulation on screen, were given a brief introduction to it, and then asked to complete a series of tasks. Clearly, this type of testing is addressing the learnability of the device but not its day-to-day usage after a period of time when users will have explored a range of facilities and either continued to use them or given up. Also, a user's learning process in the home may be assisted by members of the family or an instruction book. While the project wished to study the intuitiveness of the products, the results have to be interpreted within this context of evaluation.

9.1.4. Conclusions. The project was able, by the use of the process described, to develop a set of design rules which were employed within consumer products. Thus, the usability methods selected were successful in producing useful project results. The written reports to the companies, containing both evaluation data and recommendations for improvement, were supplemented with video clips showing key problems experienced by users as well as highlighting facilities that worked well. During the project, several rounds of evaluation were conducted as part of an iterative design cycle. This proved to be an effective means of developing the user interface design rules.

The exposure of the human-centred approach to the industrial partners enabled them to adopt similar procedures within future projects. The standard procedures documenting the evaluation processes, as well as the design rules, were distributed within the partner companies.

9.2. EVALUATION OF FINANCIAL SYSTEMS

9.2.1. Purpose. Between 1993 and 1994, HUSAT performed a number of usability evaluations of a range of systems for money-market traders and dealers, for a major financial information organization which recognized the need to improve the usability of its systems (Maguire & Carter, 1993). These evaluations were conducted at the client's site allowing them to employ representative users (traders and dealers) within the City of London. Again, due to the need for rapid feedback to design staff, these evaluations were conducted intensively—typically with one to two days of preparation, one week of testing and three days to analyse the data and report the results.

9.2.2. Selection and employment of methods. The aim was to obtain measures of performance and feedback from users, and to keep user sessions within a tight time scale. The approach adopted was a **controlled user test (7.4)** where the users were asked to perform a series of tasks, where performance data were collected and metrics calculated. User satisfaction data were also recorded using the SUMI **satisfaction questionnaire (7.5)**.

9.2.3. Constraint. A typical problem encountered was that the cooperation of users could be difficult to obtain. Due to the nature of their job, some users (e.g. stock market traders) often had to cancel their test sessions. This meant that gaps in the testing schedule arose and could only be filled by less representative personnel.

9.2.4. Conclusions. Following the successful application of this evaluation process to a number of financial products, it became company policy that all existing and future products would be evaluated formally with representative end-users.

9.3. EVALUATION OF SEARCH ENGINES

9.3.1. Purpose. A study was performed to compare a new search engine, designed to deliver information to science researchers, with two existing search engines as benchmark systems (Maguire & Phillips, 2001). The aim was to assess how well the system matched competitor systems.

9.3.2. Selection and employment of methods. It was decided to ask users to perform a **controlled user test (7.4)** of the three search engines. This was carried out by recruiting a pool of academic researchers from local universities. The study included 32 researchers from the disciplines of agriculture, biochemistry, biological science, chemical engineering, chemistry, computer science, life sciences and physics. Recruitment was carried out by advertising for participants via email through departmental distribution lists. Each volunteer was sent a document presenting the instructions to follow to test each of the search engines in turn. They self-recorded their results and responses onto the document and returned it to the experimenter. The order of usage of the search engines was randomized.

The basic test procedure for each user was to specify a query relevant to their own research, apply it to each search engine, and rate the relevance of the results. By rating the usefulness of the first 15 items returned, an interesting metric was calculated, i.e. mean relevance of items corresponding to their order in the list. Plotting these mean values for each search engine compared the relevance of items and their ordering in the list of those retrieved. Having tried all three search engines, each user was asked to place them in order of preference, based on a range of design quality criteria, and to specify which search engine was preferred in general and whether it was likely to be used for their future research. These preference data corresponded well with the comparative relevance ratings.

9.3.3. Conclusion. The testing approach of providing users with a set of instructions to follow, and for them to test the search engines in their own time, proved a convenient way to organize user trials in a short time. Requesting users to perform a task to support their own work meant that the search engines were used for real tasks. This, together with a payment in the form of book tokens, increased their motivation to take part and provide feedback quickly. The approach proved successful and was requested by the client for a follow-up evaluation of a full text database evaluation.

9.4. DEVELOPMENT OF INTRANET SITE

9.4.1. Purpose. A study was carried out to evaluate and redesign an intranet site for a police service in the UK (Maguire & Hirst, 2001*a*). The main elements of the project were the following.

- To perform user interviews with police personnel to establish user requirements for the redesign of the pages.
- To perform an expert evaluation of the current intranet pages.
- To make recommendations for redesign.

9.4.2. Selection and employment of methods. Human Factors consultants performed the study working with a police officer who was project manager for intranet development, and a civilian coordinator with a knowledge of police procedures and human factors. A series of **user requirements interviews (5.3)** were arranged and carried out with selected stakeholders to determine current perceptions of the intranet and to highlight where the current service succeeded and failed to meet stakeholder requirements. These included a constable, sergeant, inspector, non-police staff responsible for financial services, information personnel and senior officers. A semi-structured interview schedule was drawn up which covered the areas of: user needs and aspirations regarding the intranet, how well the current system meets those needs, and possible improvements that could be made. Users were given access to the intranet site so they could demonstrate their comments.

Following the stakeholder interviews, an **expert review (7.3)** of the intranet pages was performed to establish the strengths and weaknesses of the service from the point of view of general presentation and layout, usability and level of content. General recommendations for change were made following the expert evaluation. These were discussed with police representatives and different options for concept designs were proposed using **storyboards (6.4)** and **screen prototypes (6.8)**. These methods matched the requirement to create and discuss rapid prototypes within the design team. Having developed the design concept, several options for the graphic design of the site were produced as **software prototypes (6.8)** to provide both look and feel. These were discussed with the police service in several sessions. A final design for the home page was then produced and further developed to cover secondary level content pages. Web templates were then produced to allow the police service to install the new pages and to continue redevelopment in the future.

9.4.3. Conclusion. The project shows how a combination of expert evaluation, user interviews and iterative design can produce an acceptable new system design within a relatively short time (approximately 3 months).

9.5. EXPERT EVALUATION OF TRAINING OPPORTUNITIES SOFTWARE

9.5.1. Purpose. An evaluation was carried out on a website which provided information about business-related courses to small and medium enterprises (SME's) (Maguire & Hirst, 2001*b*). This was part of a programme of work to develop user requirements for a web-based e-learning service or "virtual campus".

9.5.2. Selection and employment of methods. An evaluation was performed by two experts who spent time reviewing each of the main parts of the system from their own experience, a knowledge of typical tasks and Human Factors principles **(heuristic and expert evaluation, 7.3)**. The elements included: general design and main menu, system home page, registration procedure, search facility, stored searches, career information, etc. When providing comments on the system the aim was not to improve the current system but to identify features and implications for the new system. The report also provided feedback on the current virtual campus specification to give suggestions for modification and improvement. Inputs (from a usability perspective) were made to the user specification of the new virtual campus system. These included comments such as the following.

- The system specification needs to cover course providers as well as users.
- A mechanism is needed for entering, modifying and deleting course information and course modules.
- Provide some typical scenarios of use by the users to make sure that the supplier and customer have the same "vision" of the system.

9.5.3. Conclusion. The project demonstrated how expert evaluation of a current system can provide useful feedback on the requirements specification for the new system.

9.6. EVALUATION OF ELECTRONIC PROGRAMME GUIDE (EPG)

9.6.1. Purpose. A series of studies were made by HUSAT staff to evaluate a new electronic programme guide (EPG) for a digital television service (Maguire, Graham & Hirst, 1999). The aim was to assess how usable a new digital TV system would be for typical TV viewers.

9.6.2. Selection and employment of methods. **Context-of-use analysis (4.2)** highlighted the following characteristics of usage: viewing at a distance, limited space to display information, interaction via a handset, service integrated with TV programmes, system often used on a casual basis, without use of a manual and in relaxed mode. A **software prototype (6.8)** had been developed. The system included a range of features. A "Now and Next" facility, displayed in a window, showed the name of the programme being watched on the current channel and what was coming next. The user could filter the large number of channels by type (e.g. sport, movies, children's, documentaries) or select to see TV programmes by subject. Programme details were presented either in list format (as in a newspaper) or grid format. Video recording reminders to watch selected programmes and parental control facilities were also provided.

The prototype EPG was tested using **assisted evaluation (7.2)** at the offices of the development organization. This involved setting up three simulated lounge areas so that user sessions could be run simultaneously by different evaluators. Recording equipment was brought in from HUSAT's portable laboratory. Users were recruited according to specific characteristics through a recruiting agency. Users were required to perform a series of specified tasks in a cooperative fashion with the evaluator who observed their interactions and comments. Sixteen user sessions were carried out within a 2-day period (over a weekend) when the system prototype was not being worked on by the development team.

9.6.3. Conclusion. The study demonstrated how user trials could be carried out within a simulated environment, off site, over a short period of time. It also showed how Human Factors analysts from the consultancy organization could work with a Human Factors specialist from the client organization.

9.7. INTERVIEWS TO ASSESS FUTURE REQUIREMENTS FOR FINANCIAL SERVICES

9.7.1. Purpose. A study was carried out to interview family groups to study their management of home finances (Maguire, 1999*b*). This was intended to be an information capture session to feed into future systems developments.

9.7.2. Selection and employment of methods. **Context-of-use (4.2)** information was gathered, supported by photographs taken of rooms where financial tasks were carried out. The interviewers held a series of **focus group sessions (5.4)** within each household to discuss how and where they performed financial tasks and how they would like to receive services in future and through which devices, e.g. TV, PC or other domestic appliances. The sessions were video-taped and areas and devices in the home were photographed.

9.7.3. Conclusion. The study provided useful data as a baseline for developing devices to deliver financial services in the future. It also showed how focus groups sessions could be held within family homes and provide useful contextual information about how and where family members performed current financial tasks.

10. Conclusion

The achievement of usability within system design requires a combination of the following.

(1) Careful planning of human-centred design processes.
(2) Understanding the context of use for the system as a basis for identifying requirements and evaluating the system.
(3) Understanding and specifying user requirements in a clear manner which can be assessed for achievement.
(4) System and user interface development based on a flexible and iterative approach.
(5) Usability evaluation based on both expert and user testing at appropriate points.

It is becoming increasingly obvious that a high level of usability via human-centred design is the key to future commercial success for the myriad of IT systems, consumer and PC software, internet and on-line services and telecommunications products being developed. To ensure a successful outcome, the design team must satisfy the needs and wants of the user when the development is complete. To achieve this, the users of future systems must be represented throughout the process as the best option for producing usable and successful products. Following the principles and performing the activities of ISO 13407 implements an ideal framework to ensure full representation of the users throughout the software design process.

The author is grateful to Emeritus Professor Brian Shackel (Loughborough University) for encouraging him to publish this paper, for his continued support and valuable editing role to support its production. He wishes to thank Dr Nigel Bevan (Serco Usability Services) for helpful and extensive discussions on current HCD methods and practice, and for contributing to several sections of the paper. He is grateful to Mr Jonathan Earthy (Lloyds Register) for his knowledgeable review comments and perspectives on the management of HCD within the system design process. He is also grateful to the anonymous referees for many helpful suggestions and to Kathy Phillips, HUSAT Information Scientist, for her help in locating reference details. Finally, the author would also like to acknowledge the European Commission which funded Telematics Applications Programme INUSE (IE 2016) and RESPECT (RE 2010) projects. Much of the work on documenting usability methods and relating them to the human-centred design process was performed within these projects.

References

Allabrune, A., Blandin, O., Blum, F., Clarke, A. M., Shor, O., Cocquebert, E., Debard, M., Eschenbaecher, J., Mahieu, C. & Segarra, G. (1999). *Generic Enterprise Change Management Methodology. Final Description*, 7th December 1999, p. 38. CEC ESPRIT R&D Programme Project 24771 AIT-IMPLANT Deliverable 3.2. (A CD-ROM is also available from the HUSAT Research Institute.)

ALLISON, G., CATTERALL, B., DOWD, M., GALER, M., MAGUIRE, M. & TAYLOR, B. (1992). Chapter 2: Human factors tools for designers of IT products. In M. GALER, S. HARKER & J. ZIEGLER Eds. *Methods and Tools in User-centered Design for IT* (*Human Factors in Information Technology series*, *no*. 9). Amsterdam: North-Holland.

BEVAN, N. (1999). Industry standard usability tests. In S. BREWSTER, A. CAWSEY & G. COCKTON, Eds. *Human–Computer Interaction—INTERACT*'99, Vol. II, pp. 107–108. London: British Computer Society (on behalf of IFIP).

BEVAN, N. (2001). Cost benefit analysis. *TRUMP D3.0.* London: Serco Usability Services. http://www.usability.serco.com/trump/methods/integration/costbenefits.htm.

BEVAN, N., KIRAKOWSKI, J., CLARIDGE, N., GRANLUND, Å. & STRASSER, A. (2001). UsabilityNet: designing a web site of usability resources. In J. VANDERDONCKT, A. BLANDFORD & A. DERYCKE, Eds. *Proceedings of Joint AFIHM-BCS Conference on Human–Computer Interaction IHM-HCI'2001*, Vol. 2. Toulouse: Cépaduès-Editions (in press). (see also: http://www.usabilitynet.org)

BEVAN, N. & MACLEOD, M. (1994). Usability measurement in context. *Behaviour and Information Technology*, **13,** 132–145.

BEYER, H. & HOLTZBLATT, K. (1998). *Contextual Design*: *Defining Customer-Centered Systems.* San Francisco: Morgan Kaufmann Publishers.

BIAS, R. G. & MAYHEW, D. J., Eds. (1994). *Cost Justifying Usability*. Boston: Academic Press.

BLOMBERG, J., GIACOMI, J., MOSHER, A. & SWENTON-HALL, P. (1993). Ethnographic field methods and their relation to design. In D. SCHULER & A. NAMIOKA, Eds. *Participatory Design*: *Principles and Practices.* NJ: Lawrence Erlbaum.

BOOHER, H. R. Ed. (1990). *MANPRINT*: *An Approach to Systems Integration.* New York: Van Nostrand Reinhold.

BROOKE, J. (1996). SUS: a "quick and dirty" usability scale. In P. W. JORDAN, B. THOMAS, B. A. WEERDMEESTER & I. L. MCCLELLAND, Eds. *Usability Evaluation in Industry*, pp. 189–194. London: Taylor & Francis.

BRUSEBERG, A. & MCDONAGH-PHILP, D. (2001). New product development by eliciting user experience and aspirations. *International Journal of Human-Computer Studies*, **55,** 435–452. doi:10.1006/ijhc.2001.0479.

BURMESTER, M., Ed. (1997). *Guidelines and Rules for Design of User Interfaces for Electronic Home Products.* ESPRIT 6994. Stuttgart: Fraunhofer IRB Verlag. http://www.irb.fhgde/verlag.

CAPLAN, S. (1990). Using focus group methodology for ergonomic design. *Ergonomics*, **33,** 527–533.

CARROLL, J. M., KOENEMANN-ELLIVEAU, J., ROSSON, M. B. & SINGLEY, M. K. (1993). Critical incidents and critical themes in empirical usability evaluation. In J. L. ALTY, D. DIAPER & S. GUEST, Eds. *People and Computers VIII*, pp. 279–292. Cambridge: Cambridge University Press.

CATTERALL, B. (1990). The HUFIT functionality matrix. In D. DIAPER, G. COCKTON, D. GILMORE & B. SHACKEL, Eds. *Human–Computer Interaction—INTERACT'90*, pp. 377–382. Amsterdam: North-Holland.

CHIN, J. P., DIEHL, V. & NORMAN, K. L. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. *Human Factors in Computing Systems CHI*'88, pp. 213–218. New York: ACM & Addison-Wesley.

CLARK, L. (1991). The use of scenarios by user interface designers. In D. DIAPER & N. HAMMOND, Eds. *People and Computers VI*, pp. 103–115. Cambridge: Cambridge University Press.

COOPER, A. (1999) *The Inmates are Running the Asylum*: *Why High Tech Products Drive us Crazy and How to Restore the Sanity*. Indianapolis: Sams Publishing.

DAMODARAN, L. (1996). User involvement in the systems design process—a practical guide for users. *Behaviour and Information Technology*, **15,** 262–377.

DAMODARAN, L. (1998). Development of a user-centred IT strategy: a case study. *Behaviour and Information Technology*, **17,** 127–134.

DAMODARAN, L., SIMPSON, A. & WILSON, P. (1980). *Designing Systems for People*, pp. 25–31. Manchester: NCC National Computing Centre.

DSE (1992). *UK Health and Safety Display Screen Equipment Regulations*. The Health and Safety Executive. London: HMSO.

DUMAS, J. S. & REDISH, J. C. (1993). *A Practical Guide to Usability Testing*. Norwood, NJ: Ablex Publishing Corp.

EARTHY, J., SHERWOOD JONES, B. & BEVAN, N. (2001). The improvement of human-centred processes—facing the challenge and reaping the benefit of ISO 13407. *International Journal of Human–Computer Studies*, **55,** 553–585. doi:10.1006/ijhc.2001.0493.

EASON, K. D. (1984). Towards the experimental study of usability. *Behaviour and Information Technology*, **3,** 133–143.

EASON, K. D. (1988). *Information Technology and Organisational Change*. London: Taylor & Francis.

EASON, K. D. & OLPHERT, C. W. (1995). Early evaluation of the organisational implications of CSCW systems. In P. J. THOMAS, Ed. *CSCW Requirements and Evaluation*, pp. 75–89. London: Springer-Verlag.

ERICSSON INFOCOM CONSULTANTS AB AND LINKÖPING UNIVERSITY (2001). The Delta method. www.deltamethod.net/

FARLEY, J., Ed. (1997). *The Focus Group: A Strategic Guide to Organizing, Conducting and Analyzing the Focus Group Interview (2nd edn)*. Templeton: Probus Publishing.

FITTER, M., ROBINSON, D., RECTOR, A., HORAN, B., NOWLAN, A., WILSON, A. & NEWTON, P. (1991). *Supportive Evaluation Handbook: a Methodology to Facilitate System Development in the Health Care Domain*. Social & Applied Psychology Unit, Sheffield University, Sheffield.

FLANAGAN, J. C. (1954). The critical incident technique. *Psychological Bulletin*, **51,** 28–35.

FOWLER, C. (1991). Usability Evaluation—usability in the product lifecycle. *Usability Now! Newsletter*, Issue 3, Spring, pp. 6–7. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

GALDO, E. M. DEL, WILLIGES, R. C. & WILLIGES, B. H. (1986). An evaluation of critical incidents for software documentation design. In *Proceedings of the Human Factors Society 30th Annual Meeting, Dayton, OH, 29 September–3 October*, pp. 19–23. The Human Factors and Ergonomics Society, P.O. Box 1369, Santa Monica, CA 90406-1369, USA.

GALE, A. & CHRISTIE, B. (1987). *Psychophysiology and the Electronic Workplace*, pp. 329–332. Chichester: Wiley.

GOULD, J. D., CONTI, J. & HOVANYECZ, T. (1983). Composing letters with a simulated listening typewriter. *Communications of the ACM*, **26,** 295–308.

HACKOS, J. & REDISH, J. (1998). *User and Task Analysis for Interface Design*. Chichester: Wiley.

HALL, R. (2001). Prototyping for usability of new technology. *International Journal of Human-Computer Studies*, **55,** 485–501. doi:10.1006/ijhc.2001.0478.

IP, W. K., DAMODARAN, L., OLPHERT, C. W. & MAGUIRE, M. C. (1990). The use of task allocation charts in system design—a critical appraisal. In D. DIAPER, G. COCKTON, D. GILMORE & B. SHACKEL, Eds. *Human-Computer Interaction INTERACT'90*, pp. 289–294. Amsterdam: North-Holland.

ISO (1997*a*). *ISO 9241: Ergonomics Requirements for Office Work with Visual Display Terminals (VDTs)*, 17 parts. Geneva: International Standards Organisation. Also available from the British Standards Institute, London.

ISO (1997*b*). *ISO 9241-11: Ergonomic requirements for Office Work with Visual Display Terminals (VDTs). Part* 11—*Guidelines for Specifying and Measuring Usability*. Geneva: International Standards Organisation. Also available from the British Standards Institute, London.

ISO (1999). *ISO 13407: Human-centred Design Processes for Interactive Systems*. Geneva: International Standards Organisation. Also available from the British Standards Institute, London.

ISO (2000*a*). *ISO TR 18529: Ergonomics of Human System Interaction—Human-centred Lifecycle Process Descriptions*. Geneva: International Standards Organisation. Also available from the British Standards Institute, London.

ISO (2000*b*). In M. AZUMA, N. BEVAN & W. SURYN, Eds. *ISO/IEC DTR* 9126-4: *Software Engineering—Software Product Quality—Part 4: Quality in Use Metrics*. Committee: ISO/IEC JTC 1/SC 7/WG 6. Geneva: International Standards Organisation, 12 November 2000.

Jones, J. C. (1980). *Design Methods: Seeds of Human Futures*. Chichester: Wiley.

Kirakowski, J. (1996). The software usability measurement inventory: background and usage. In P. W. Jordan, B. Thomas, B. A. Weerdmeester & I. L. McClelland, Eds. *Usability Evaluation in Industry*, pp. 169–178. London: Taylor & Francis.

Kirakowski, J. & Claridge, N. (2001). WAMMI—website evaluation tool. See: *http://www.wammi.com.*

Kirwan, B. & Ainsworth, L. K., Eds. (1992). *A Guide to Task Analysis*. London: Taylor & Francis.

Lindgaard, G. (1994). *Usability Testing and System Evaluation—a Guide for Designing Useful Computer Systems*. London: Chapman & Hall Computing Series.

Macaulay, L.A. (1996). *Requirements Engineering*. Berlin: Springer-Verlag Series on Applied Computing.

Madsen, K. & Aiken, P. (1993). Experiences using co-operative interactive storyboard prototyping. *Communications of the ACM*, **36,** 57–64.

Maguire, M. C. (1996). *Prototyping and Evaluation Guide*. EC Telematics Applications Programme document, Project IE 2016 INUSE. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

Maguire, M. C. (1998). *User-Centred Requirements Handbook*. EC Telematics Applications Programme, Project TE 2010 RESPECT (Requirements Engineering and Specification in Telematics), WP4 Deliverable D4.2, Version 3.3, May 1998. (http:www.lboro.ac.uk/research/husat/respect/rp2.html).

Maguire, M. C. (1999*a*). A review of user-interface design guidelines for public information kiosk systems. *International Journal of Human-Computer Studies*, **50,** 263–286. doi:10.1006/ijhc.1998.0243.

Maguire, M. C. (1999*b*). *NCR Knowledge Lab. Report on study of the management of domestic finances by family groups*, 7 *May* 1999. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

Maguire, M. C. (2001*a*). Context of use within usability activities. *International Journal of Human–Computer Studies*, **55,** 453–483. doi:10.1006/ijhc.2001.0486.

Maguire, M. C. (2001*b*). *TAQ and SAQ: Pre and post test questionnaires for assessing user acceptance*. HUSAT Memo HM 1148, June 2001. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

Maguire, M. C. & Carter, A. (1993). *Evaluation of Reuters RACE 3.0 Prototype*, *11 November 1993*. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

Maguire, M. C., Graham, R. & Hirst, S. J. (1999). *Digital TV electronic programme guide comparative study—Phase* 3 *Report on user trials and focus groups*, *12 March 1999.* HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

Maguire, M. C. & Hirst, S. (2001*a*). *MPS Redesign of corporate intranet pages. HUSAT Consultancy Limited*, *23 March 2001*. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

Maguire, M. C. & Hirst, S. (2001*b*). *Usability evaluation of the LINK project TIGS website and feedback on OVC specification. HUSAT Consultancy Limited*, *2 March 2001.* HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

Maguire, M. C. & Phillips, K. (2001). *User testing of search engines for science information. Project number: HC/ES/220, April 2001*. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

Maguire, M. C. & Vaughan, G. (1997). *Introducing New Technol*ogy. INUSE Project Report, May 1997. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

Maissel, J., Dillon, A., Maguire, M. C., Rengger, R. & Sweeney, M. (1991). *Context Guidelines Handbook*. MUSiC project deliverable IF2.2.2. HUSAT Research Institute, Loughborough, Leicestershire, UK and National Physical Laboratory, Teddington, UK.

MAULSBY, D., GREENBERG, S. & MANDER, R. (1993). Prototyping an intelligent agent through Wizard of Oz. In S. ASHLUND, K. MULLET, A. HENDERSON, E. HOLLNAGEL & T. WHITE, Eds. *Human Factors in Computing Systems—INTERCHI'93*, pp. 277–284. ACM & IFIP TC.13. New York: Addison-Wesley. Amsterdam: IOS Press.

MCDONALD, J. E. & SCHVANEVELDT, R. W. (1988). The application of user knowledge to interface design. In R. GUINDON, Ed. *Cognitive Science and its Applications for Human–Computer Interaction*, pp. 289–338. Hillsdale: Lawrence Erlbaum.

MOD (2000). *Human Factors Integration—An Introductory Guide, Version 1.2 dated 16 August 2000*. UK: Ministry of Defence.

MONK, A., WRIGHT, P., HABER, J. & DAVENPORT, L. (1993). *Improving Your Human–Computer Interface: A Practical Technique*. London: Prentice-Hall International (UK) Ltd.

NICOLLE, C. & ABASCAL, J. (2001). *Inclusive Design Guidelines for HCI*. London: Taylor & Francis.

NIELSEN, J. (1991). Paper versus computer implementations as mock up scenarios for heuristic evaluation. In D. DIAPER, G. COCKTON, D. GILMORE & B. SHACKEL, Eds. *Human-Computer Interaction—INTERACT'90*, pp. 315–320. Amsterdam: North-Holland.

NIELSEN, J. (1992). Finding usability problems through heuristic evaluation. In P. BAUERSFELD, J. BENNETT & G. LYNCH, Eds. *Human Factors in Computing Systems—CHI'92*, pp. 373–380. New York: ACM & Addison-Wesley.

NIELSEN, J. (1993). *Usability Engineering*. London: Academic Press.

NIELSEN, J. (1994). Special Issue: usability laboratories. *Behaviour and Information Technology*, **13,** 3–197.

NIELSEN, J. (2000*a*). The use and misuse of focus groups. http://www.useit.com/papers.

NIELSEN, J. (2000*b*). *Designing Web Usability: The Practice of Simplicity*. Indianapolis: New Riders Publishing.

NIELSEN, J. & LANDAUER, T. K. (1993). A mathematical model of the finding of usability problems. In S. ASHLUND, K. MULLET, A. HENDERSON, E. HOLLNAGEL & T. WHITE, Eds. *Human Factors in Computing Systems—INTERCHI'93*, pp. 206–213. ACM & IFIP TC.13. New York: Addison-Wesley. Amsterdam: IOS Press.

OLPHERT, C. W. & DAMODARAN, L. (1991). Job design, work organisation and system design. Case studies in information technology. In K. LEGGE, C. CLEGG & N. KEMP, Eds. *People and Organisations*, pp. 273–287. Manchester: NCC Blackwell Ltd.

OLPHERT, C. W. & HARKER, S. D. P. (1994). The ORDIT method for organisational requirements definition. In G. E. BRADLEY & H. W. HENDRICK, Eds. *Human Factors in Organizational Design and Management—IV*, pp. 421–426. Amsterdam: North Holland.

OSBORN, A. F. (1963). *Applied Imagination*. New York: Schribeners and Sons.

POULSON, D., ASHBY, M. & RICHARDSON, S., Eds. (1996). *USERfit A Practical Handbook on User-centred Design for Assistive Technology*. Handbook produced within the European Commission TIDE programme USER project. HUSAT Research Institute, The Elms, Elms Grove, Loughborough, Leicestershire LE11 1RG, UK.

PREECE, J., ROGERS, Y., SHARP, H., BENYON, D., HOLLAND, S. & CAREY, T. (1994). *Human–Computer Interaction*. Reading, MA: Addison-Wesley.

PRUE (2001). http://www.usability.serco.com/prue.

RETTIG, M. (1994). Prototyping for tiny fingers. *Communications of the ACM*, **37,** 21–27.

ROBERTSON, S. & ROBERTSON, R. (1999). *Mastering the Requirements Process*. Harlow, UK: Addison-Wesley.

ROBERTSON, S. (2001). Requirements trawling: techniques for discovering requirements. *International Journal of Human–Computer Studies*, **55,** 405–421, doi:10.1006/ijhc.2001.0481.

SAZEGARI, S., ROHN, J. A., UYEDA, K. M., NEUGEBAURER, C. & SPIELMANN, N. (1994). Building a usability laboratory (Section with 4 papers). *Behaviour and Information Technology*, **13,** 20–50.

SHACKEL, B. (1981; 1984). The concept of usability. In *Proceedings of IBM Software and Information Usability Symposium*, Poughkeepsie, NY, 15–18 September 1981, pp. 1–30. Also in J. L. BENNETT, *et al.*, Eds. *Visual Display Terminals: Usability Issues and Health Concerns*, pp. 45–88. Englewood Cliffs, NJ: Prentice-Hall.

SHACKEL, B. (1991). Usability—context, framework, definition, design and evaluation. In B. SHACKEL & S. J. RICHARDSON, Eds. *Human Factors for Informatics Usability*, pp. 21–37. Cambridge: Cambridge University Press.

SHEPHERD, A. (1985). Hierarchical task analysis and training decisions. *Programmed Learning and Educational Technology*, **22**, 162–176.

SHEPHERD, A. (1989). Analysis and training in information technology tasks. In D. DIAPER, Ed. *Task Analysis for Human–Computer Interaction*, pp. 15–55. Chichester: Ellis Horwood.

SHNEIDERMAN, B. (1998). *Designing the User Interface: Strategies for Effective Human–Computer Interaction* (3rd edn). Reading, MA: Addison-Wesley.

SPOOL, J. & SCHROEDER, W. (2001). Testing web sites: five users is nowhere near enough. *CHI* 2001 *Extended Abstracts*, pp. 285–286. New York: ACM.

TAYLOR, B. (1990). The HUFIT Planning, Analysis and Specification Toolset. In D. DIAPER, G. COCKTON, D. GILMORE & B. SHACKEL, Eds. *Human–Computer Interaction—INTERACT'90*, pp. 371–376. Amsterdam: North-Holland.

THOMAS, C. & BEVAN, N. (1995). *Usability Context Analysis: a Practical Guide*. National Physical Laboratory, Teddington, Middlesex, TW11 0LW, UK. (Now available from N. Bevan, Serco Usability Services, 22 Hand Court, London WC1V 6JF).

VAN SCHAIK, P. (1999). Involving users in the specification of functionality using scenarios and model-based evaluation. *Behaviour and Information Technology*, **18,** 455–466.

VAUGHAN, G. M., SIEMIENIUCH, C. E., PARKER, C. G., MAGUIRE, M. C., SEGARRA, G., MSAFF, O., SOUSSI, S., DORIRY, G., COCQUEBERT, E. & MAHIEU, C. (1988). *State of the Art Enterprise Change Management*, p. 203. CEC ESPRIT R&D Programme Project 24771 AIT-IMPLANT Deliverable D2.1.

VERTELNEY, L. (1989). Using video to prototype user interfaces. *ACM SIGCHI Bulletin*, **21,** 57–61.

WHITESIDE, J., BENNETT, J. & HOLTZBLATT, K. (1988). Usability engineering: our experience and evolution. In M. HELANDER, Ed. *Handbook of Human–Computer Interaction*, pp. 791–817. Amsterdam: Elsevier.

WILSON, J. & ROSENBERG, D. (1988). Rapid prototyping for user interface design. In M. HELANDER, Ed. *Handbook of Human–Computer Interaction*, pp. 859–875. Amsterdam: North-Holland.

YOUNG, E. & GREENLEE, R. (1992). Participatory video prototyping. *Human Factors in Computing Systems—CHI'92* (Poster exhibit no. 24).