

2001

Towards Universal Accessibility: Including Users with a Disability in the Design Process

Susan Keller

Deakin University, susank@deakin.edu.au

Richard Braithwaite

Deakin University

Janet Owens

Deakin University

Kaye Smith

Deakin University

Follow this and additional works at: <http://aisel.aisnet.org/acis2001>

Recommended Citation

Keller, Susan; Braithwaite, Richard; Owens, Janet; and Smith, Kaye, "Towards Universal Accessibility: Including Users with a Disability in the Design Process" (2001). *ACIS 2001 Proceedings*. 51.

<http://aisel.aisnet.org/acis2001/51>

This material is brought to you by the Australasian (ACIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ACIS 2001 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Towards Universal Accessibility: Including Users with a Disability in the Design Process

Susan Keller^a, Richard Braithwaite^a, Janet Owens^b and Kaye Smith^b

^aSchool of Management Information Systems
Deakin University, Melbourne, Australia
susank@deakin.edu.au

^bInstitute of Disability Studies
Deakin University, Melbourne, Australia

Abstract

While increased awareness of disability issues has resulted in the development of guidelines for developing accessible software, such guidelines do not guarantee that the end product will be optimal for users with a disability. We present an overview of a user-centred approach to the needs analysis and ultimate design of a disability aware email client (Multimail). We discuss the process we used to work with participants in developing the software and reflect on the benefits and challenges of the process.

Keywords

Client centered design, accessibility, electronic mail

INTRODUCTION

Technological developments and paradigm shifts over the last four decades have changed not only how we plan, design and develop software but also how we use software. While software was once the domain of specialists and trained users, increasingly, everyone is expected to use software and technology. This means that we must design and develop software that is responsive to the needs of a diverse range of users, including those with disabilities.

In many countries, legislation, such as Australia's Disability Discrimination Act, provides an additional incentive for designing software that is accessible for everyone, including those with disabilities. This was highlighted in a recent court case in Australia which upheld a blind man's right to access the content on the Olympic Web site (QC, 2000).

Despite the intentions behind the legal imperative, ensuring that software is responsive to the needs of diverse users is not a trivial problem. From a systems development perspective there are two distinct issues: building the 'right' systems to ensure that important user requirements are identified and addressed; and ensuring that users with varying abilities can use them. To address the first issue we argue that users with disabilities need to be involved in the system development process, and to address the second, that developers need to familiarise themselves with the well developed accessibility guidelines that are readily available.

Unfortunately, for many people who have a disability, exclusion is the outcome of the design of many current systems. Although many software designers are unaware of accessibility issues, such exclusionary practices can be regarded as discriminatory. The Vocational Rehabilitation Branch of the International Labour Office (1999) has argued that discriminatory practices restrict the equal opportunities of people who have a disability. For people who have a disability, system design can determine whether the technology is a powerful proponent or a major barrier in ensuring equality of opportunity. Including people with disabilities in the system design and development process facilitates the creation of universally accessible and usable products. If technology is to respond to a diverse range of needs, inclusion of people who have a disability needs to be more than a token gesture. Such inclusion needs to ensure wide representation of potential users.

In this paper we report on a research project which aimed to first investigate the on-line communication requirements of computer users with disabilities and then to produce an accessible email client. While the goals of universal accessibility are well reported, there is a paucity of literature on requirements engineering and system development methodologies that promote it. In particular, and the motivation for writing this paper, we were unable to find any literature on the most practical means for including people with disabilities in the design process. In completing the research project we gained many insights into the challenges of accessibility from a systems development perspective as well as the importance and challenges of including users with a disability in the design process. In this paper we:

- give an overview of the user-centred system development process we used
- detail how user requirements and needs were translated into an accessible email application
- and finally we reflect on the process highlighting benefits and challenges and offer suggestions for future research

BACKGROUND

Disability can cover a range of impairments including physical, sensory and cognitive, all of which can impact on a person's ability to interact with computer technology. There has been a shift in the last decade or so in the way disability, as a phenomenon has been understood. This shift has involved an increasing emphasis on the way disability is experienced, or the consequences of disability in a particular environment. Roulstone (1998) argued that a person is disabled to the extent that he/she is prevented from full participation in society by constructed barriers. For example, lack of wheel chair access is a barrier that prevents a wheel chair user from enjoying a concert. Similarly, lack of keyboard control of an application is a barrier that prevents a person who cannot use a mouse from using the software. While using the World Wide Web, a visually impaired user could be prevented from purchasing goods on line if a site presents essential information in the form of graphics rather than text that could be fed to a speech engine. From this perspective, we as information technologists have an important role in facilitating the full participation of individuals in the information economy.

The impetus for addressing the issue of accessibility becomes increasingly urgent when you consider the proliferation of information systems (including the Web) in our everyday life. Increasingly, we are encouraged and often expected to use technology for obtaining information, banking, recreation, government services, education and shopping services (NOIE 2000, EU 2000). This trend is very likely to continue to the point where an individual will be at a severe disadvantage (socially, intellectually and economically) if he or she cannot access on-line services (Achieving Universal Access 2000). As the Web continues to evolve and moves further away from text-based delivery, universal accessibility becomes increasingly complex as a whole new wave of technical and design issues must be contended with.

The number of people who experience some type of disability is significant. According to Vanderheiden (1994) approximately 18 to 20% of any population has a disability and as we age our chances of acquiring some form of functional limitation increases dramatically. By age sixty-five, 50% of people will experience some functional limitation and by the age of seventy this rises to 75%. People are living longer and the number of elderly people as a percentage of the total population is steadily rising in many parts of the world. For example, in Australia it is estimated by the year 2051, between 24 and 27% of the population will be aged 65 or older (ABS 2000a, ABS 2000b).

Approaches to accessible software systems

Access to technology has several dimensions including: availability, continuity, affordability, accessibility and awareness. The accessibility dimension is concerned with building information technology hardware, software and services in such a way that they do not create barriers and exclude people from their use (EC 1997). There are four basic approaches to providing accessible software systems:

- develop specifically for a particular disability group
- rely on operating system accessibility features
- employ third party assistive technology
- design for universal accessibility

These approaches are not mutually exclusive. Two or more may be required in order to ensure accessibility.

Develop Specifically for a Particular Disability Group.

This approach involves designing and implementing software or services specifically to suit the needs of a particular disability group. An example of this is *pwWebSpeak*, a Web browser designed especially for users with impaired vision. This approach to development has the potential to provide the best possible solutions for specific groups because the design is tailored to suit the needs and abilities of the target group. There are also, however, significant disadvantages to this approach. Software produced this way is suited to a relatively small target group hence such products tend to be expensive and thus potentially unaffordable. Another problem is that such specificity is unlikely to cater well for users with multiple disabilities. For example, with conditions such as Multiple Sclerosis it is possible that a user may have both a visual impairment and a motor impairment.

Use of Operating System Accessibility Features.

Accessibility features that are built into the operating system are becoming increasingly sophisticated. Microsoft began to offer accessibility options as part of Windows 95 with features such as high contrast mode, sticky keys, slow keys, bounce keys and sound sentry and later versions of Windows have extended the range of in-built accessibility options. Such operating system features mean that many more people can use 'standard' software than would otherwise be the case. This has obvious advantages in terms of availability and cost, however, software may not be very useable even if it is accessible.

Third Party Assistive Technology.

Third party assistive technologies are software and hardware products that work in conjunction with both the operating system and the application to provide specialist input and output services. Common examples include screen readers, switch devices with scanning software and alternative keyboards. Depending on the devices in question, assistive technologies may be useful with standard packages or may require speciality packages.

Universal Accessibility.

Designing systems that are useable by the broadest range of people operating in the widest range of conditions is often referred to as universal design, universal accessibility (Story, 1998) (Stephanidis and Emiliani, 1999) or universal useability (Vanderheiden, 2000). It is important to note that this does not imply designing systems just for people with disabilities, but designing systems to accommodate a diverse range of needs including those presented by people who have a disability. Disability is only one variation in human performance that universal usability attempts to address. Vanderheiden (2000) argued that the benefits of universal design are not exclusive to people who have a disability. For example, although a person who is blind needs to operate a system without sight, the same requirements for sightless operation are also relevant for people operating a system while driving a car or working in the dark. Thus universal accessibility is concerned with "designing application user interfaces that are easier to use for users with disabilities as well as users 'without' disabilities by taking their needs into account when system and application software is designed" (Bergman and Johnson 1997).

However, universal accessibility is not an easy goal. In the Universal Design Research Project (1998) an initial survey of twenty-two technology companies found there were many barriers to the implementation of universal design. These included fear of costs in retooling and retraining and fear of slowing down time to market. Other barriers identified were lack of interest including the belief that universal design was simply designing for disability and therefore not a target market. Organisational structures that make changes to design methods difficult were also highlighted as a barrier. The same study pinpointed enablers including a belief that universal design would increase market share and that it could be cost effective. Internal champions, top level management support, regulatory demands as well as knowledge of the issues and possible solutions were also seen as motivators. Interestingly, product testing that included consumers from diverse groups (including those with disabilities) was suggested as a strategy for facilitating the adoption of universal design.

However, if truly useful products are to be developed then consumers should be involved in the design stage and not just at the testing stage. The importance of gaining this perspective can be seen in the research conducted by Sinks and King (1998) which highlighted the security fears many people with disabilities have about using e-commerce sites. The issue is not access to the site but rather fear of giving out delivery information that could potentially threaten a person's physical safety. Until such needs are addressed it can be predicted that a section of the community who could benefit from e-commerce will not use it.

Stephanidis goes further contending that universal accessibility should be "a first-order design objective and a compulsory quality target in the emerging Information Society" (Stephanidis 1999).

THE SOFTWARE DESIGN AND DEVELOPMENT PROCESS

The MultiMail Project

This paper describes the second phase of a two-part project aimed at designing and developing an accessible email client. The first phase investigated the on-line communication requirements of users with a disability to determine barriers to successful use of email and the extent to which existing assistive technology addresses these difficulties.

The methodology and results of the first phase of the project have already been published (Keller et al. 2000). In the second phase of the project we employed a user-centred approach to the design and development of an accessible email application called MultiMail. The MultiMail project also built on the findings of the MultiWeb project (Owens and Keller 2000) and LEAD project (Braithwaite and Baxter 1997).

Overview of the System Development Process

Participatory design approaches are well established in the literature but are usually focused on empowering workers in their jobs (Macaulay, 1996). However, with the MultiMail project we were aiming at developing a generic product for a large number of consumers with a very broad range of individual abilities. Rapid prototyping was considered unsuitable because many participants had health issues that would make the required commitment difficult and also because of the complexity of the access requirements that needed to be incorporated into the design. The user-centred design process as embodied in the standard ISO 13407 (Introduction to ISO 13407 1999) is useful because it suggests a broad approach to user centered design that does not necessarily imply an organisational context and also because it includes a planning phase. Figure 1 shows the five basic steps in the ISO 13407 model. The first step deals with the planning involved in including users in the development process. The next four stages suggest an iterative approach to specifying the context of use and user requirements and producing and evaluating design solutions. The techniques we used for each step in the process are shown in italics in each of the five stages (see Figure 1).

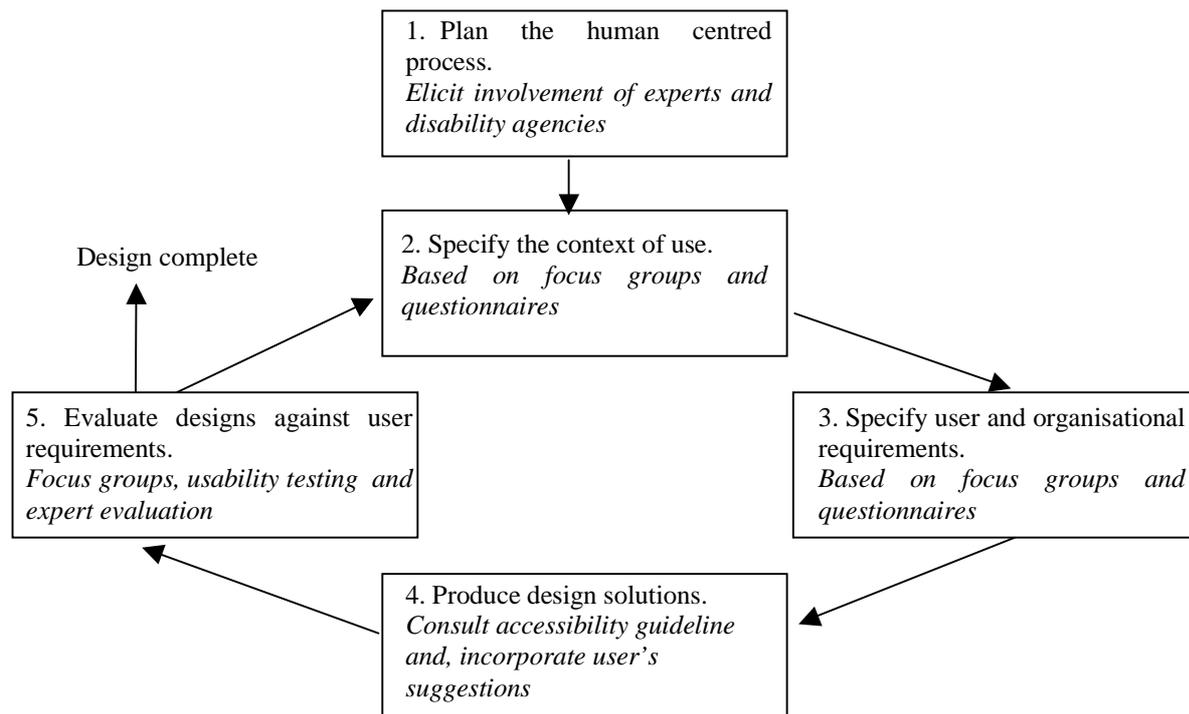


Figure 1: Application of the ISO13407 model to the design of Multimail

Stage 1: Planning the Human Centred Process

At this stage we considered it important to make explicit our view that the system development was to be driven by users with a disability who would actively be involved in making decisions as to the scope of the new system. We planned to employ some of the techniques of participatory design (focus groups, brainstorming and prototyping) as well as other data collection techniques such as survey questionnaires and usability testing. Participants in the study consisted of people with disabilities and representatives of peak disability groups and agencies within Victoria, NSW and SA. Researchers included people from the disability studies field as well as information systems field. At this stage we found co-operation was important, not just between researchers and participants, but between representatives of different disability groups.

Stage 2: Specifying the Context of Use.

The aim of specifying the context is to identify key characteristics of intended users, key tasks and essential environmental factors such as the physical and social environments in which the product is used. Key issues that emerged during this stage were the diversity of characteristics of intended users and the diversity of other stake holders (disability agencies). Potential users included those with vision impairments, physical disabilities, cognitive disabilities, hearing impairments and people with multiple disabilities. The initial round of focus groups included participants, carers and the project team.

Relevant social and environmental factors included the reality that many users with a disability have limited employment opportunities and thus low incomes. Cost of accessible computer hardware and software is a

known barrier to use of computers so we planned to develop a product with accessibility features such as scanning for switch device users and speech synthesis built-in rather than needing to be purchased as separate add-ons. One constraint mandated by the funding body was that the system had to operate on low budget equipment and consequently be compatible with all versions of Microsoft Windows from version 3.1 onwards.

Stage3: Specifying User and Organisational Requirements.

This step was where the functional requirements for MultiMail were developed. During the focus groups a very rudimentary prototype consisting of various screen layouts was demonstrated. The purpose of this was to generate discussion and to help participants envisage use of the intended product. The focus groups highlighted many broad issues such as the literacy problems and the need for keeping things simple as well as agreement on broad requirements. The output from the focus group meetings included both system requirements and user accessibility requirements. Here we need to make a distinction between system requirements in the problem domain and user accessibility requirements. While it was possible to gain broad consensus on system requirements, it was not possible to gain consensus on accessibility requirements as individuals' needs were often conflicting. Despite this, group problem solving led to many useful design suggestions which were incorporated into the design at stage 4.

The focus groups were typically three hours in duration and proceedings were audio-taped for later transcription. Focus group data was analysed using standard inductive techniques and thematic analysis. Specific suggestions made by individuals were included verbatim. An on-line survey was provided for people who were unable to participate in focus groups.

Stage 4: Producing Design Solutions.

At this stage the development team analysed participant requirements, design suggestions, and accessibility requirements and then determined feasibility in terms of project constraints. This process was extremely challenging because of the sheer diversity of accessibility needs. Accessibility guidelines were extensively consulted during this phase (Vanderheiden 1992, Vanderheiden 1994, Vanderheiden and Vanderheiden 1992, W3C 1999). These guidelines provide non-context specific access information which has been compiled in consultation with users with a disability. An example of a specific guideline is "avoid timed responses" so that a user is not limited to a certain amount of time to read information or respond to a system. Such issues are important but may not necessarily be brought out during interactions with users.

Design suggestions resulting from group brain-storming by participants at the focus groups were included in the design where this was technically feasible. These included the following:

- a simple interface with only the most basic options available. This was considered important for people with cognitive disabilities as well as those who were new to email.
- situating newly arrived messages at the top of the email message list. This was suggested as a means of eliminating unnecessary scrolling (difficult for people with physical disabilities). It also made more sense for those with cognitive disabilities.
- providing a list of generic and frequently used contacts. Participants suggested this as a way to overcome the problem of a person having no initial contacts.
- a facility to save messages as templates for future use. The ability to reuse email was considered important for people with literacy problems and for those with physical disabilities.
- providing a set of ready made messages on basic topics such as birthday greetings and meeting details . Again this was suggested to help those with cognitive and physical disabilities.

Participants also wanted word-prediction software incorporated into the design to aid literacy difficulties and touch screen and switch device users requested various on-screen keyboard layouts.

It was clear that it would be impossible to arrive at a single screen design that would meet the needs of all users and indeed, Stephanidis (1999) argues that universal design need not imply a single user interface. However, we did aim to find a design that could be used as a basis for the required variations.

As a way of analysing the accessibility requirements obtained from focus groups and the accessibility guidelines, requirements were grouped around how a user interacts with the system in terms of input and output. For example, many users can only use a keyboard as they do not have the fine motor control required for mouse operation hence every aspect of the program needs to be accessible via keyboard. Others can use a mouse if the cursor and the target area are sufficiently large. For some, the only possible input is via a switch device; a hardware input device that can respond like a single mouse click. Switch devices differ in design depending on the part of the body that is used to activate them (for example, foot, finger, head) and work in conjunction with software that allows parts of the interface to be scanned by sequentially highlighting them. When the scanning reaches the appropriate screen element the user operates the switch to activate that function. This method can be

very slow and frustrating for text entry which is accomplished via an on-screen scanning keyboard. However, for users who are unable to use other input devices (including speech input) this may be the only available option. With touch screen access, some people require the edge of the computer screen for support and have limited physical access to the centre of the screen.

Not every one can read the output of a computer system. Some people need information spoken out either because of literacy difficulties or vision impairments. Screen readers or in-built speech synthesis programs can address this requirement. With some forms of dyslexia, users need a guide to help them keep place in text. An option such as allowing the person to control a highlighted line is one possible solution.

The most demanding design challenge was if finding a workable design for touch screen and switch device users. Touch screen design is challenging because the target area needs to be large and preferably at the edge of the screen. Designing for effective switch device use is also challenging because every function must be available via scanning including scrolling, navigating, editing etc.

To accommodate these requirements we opted for a basic design based on buttons rather than menus (see Figures 2 and 3). Buttons have a number of significant advantages for accessibility: buttons provide a large target area for touch screen users and mouse users, buttons facilitate scanning by allowing buttons to sequentially receive focus, buttons also allow for large captions to be used for those with vision impairments and buttons allow controls to be positioned at the top, bottom, right or left of the screen.

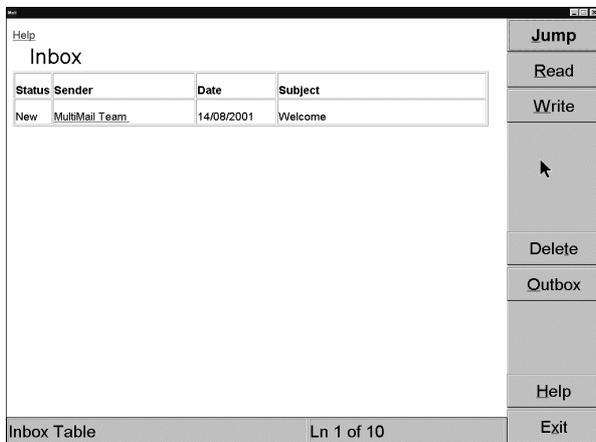


Figure 2: Simple Interface showing Inbox

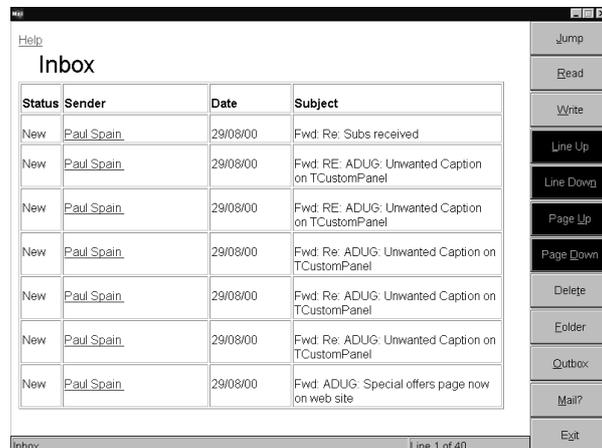


Figure 3: Switch Device Interface

To allow users to configure the software to meet individual requirements a User Options program was developed. This separate program allows a user to select the required interface such as simple or touch screen as well as options such as speech synthesis or large print.

Figure 2 shows the final design of the simple interface that participants suggested. The simple interface also includes a three part wizard approach to composing an email (not shown). Figure 3 shows the switch device interface with block scanning.

Stage 5: Evaluating the Design Against User Requirements.

During application development the prototype design was evaluated at another focus group with disability agency representatives. Representatives were positive about the design of the front end and positive comments were received about the block scanning and the concept of completing email tasks step by step. A number of useful refinements were suggested including ensuring that words in the word prediction list could be read as the user scrolled through the list. Suggestions were subsequently incorporated into the design where this was technically feasible.

When the initial software development was complete, usability testing was conducted using standard cognitive walkthrough techniques (Preece 1999). A total of 18 people evaluated interfaces, with 13 evaluating the default interface and 11 evaluating the switch device interface. Participants completed reading and writing tasks using one or more of MultiMail's interfaces. All participants completed the evaluation process for two interfaces in approximately one hour and fifteen minutes. This process resulted in some further suggested improvements such as allowing the user the option of turning off the password protection feature, providing a quick way to exit scanning mode in the switch device interface and some minor wording changes to commands. As a final iteration of the development cycle, these suggested revisions were incorporated into the design.

A summary of how MultiMail meets the access needs of computer users with particular disabilities is summarised in Table 1 below (categories are taken from Vanderheiden, 1994):

<i>For People with Physical Disabilities</i>
The program operates without the need to respond to timed responses
There is flexibility in choosing the response time for switch device and scanning
There is flexibility in choice of access, e.g. Mouse, Keyboard, Switch Device, Touch Screen
Block scanning is provided for switch device users along with a choice of 4 different on-screen keyboards.
Word prediction is provided to enhance text production
Fatigue is offset through the reuse of template messages and the ability to save messages from one sitting to another
<i>For People Who are Deaf or Hard of Hearing</i>
All information is provided visually regardless of the use of 'Speech' and 'Talking Buttons' options
Program operation does not require sound
Help information is in plain English
Product support is available at Deakin University through email contact or by telephone contact using the National Relay Service
<i>For People Who have Color Blindness</i>
Colour choices are available for all screen components (screen, button bar, buttons, text)
Colours are available which differ in darkness, thereby increasing contrast
The program can operate in a monochrome mode (e.g. default colour schemes of black on white or black on grey)
<i>For People with Low Vision</i>
Type of font, font size, and colours can be adjusted in the User Options program to make them more visible
A Large Print option automatically gives size 24 font and large cursors
There is a predictable layout for screens, navigational boxes and button bars within the program; users make their preferred choices in the User Options program
All screens, button bars and navigational boxes have a plain (non-patterned) background
'Speech' and 'Talking Buttons' options are available
<i>For People who are Blind</i>
There is a predictable layout for screens, navigational boxes and button bars within the program; users make their preferred choices in the User Options program
'Speech' and 'Talking Buttons' are available. Information on buttons and on screens can be read out; information in text boxes is spelled as it is typed
Use of buttons means that Menus do not need to be utilised
Text information is provided for pictures used in Help Information
<i>For People With Language or Cognitive Disabilities</i>
The 'Simple' interface is an easy way to send and receive mail. The Simple interface uses a simple and uncluttered step-by-step screen layout.
All messages and alerts stay on the screen until they are cancelled or accepted; these messages are read out if the 'Speech' option is chosen
All screen layouts are consistent
Spell checking is provided
Word prediction options are available to support literacy
'Speech' and 'Talking Buttons' are available. Information on buttons and on screens can be read out; information in text boxes is spelled as it is typed

Table 1: Summary of accessibility considerations for people with disabilities.

CONCLUSIONS AND FURTHER RESEARCH

There are well developed, if not well known, guidelines for developing accessible application software. However, accessibility guidelines can only provide general non-contextual information on how to make applications more accessible. The guidelines provide us with the 'build it right' information. The input of people with disabilities is vital for information on building the right system. Accessibility has many dimensions. As operating systems and assistive technologies tackle many of the technical issues it becomes increasingly important to look at building the right software for users with a disability. MultiMail offers a wide range of accessibility features (see Table 1), however, the email package design also has a number of features that are

significantly different from standard email packages. Features such as the simple to understand interface, standard built-in email messages, an easy way to save and retrieve draft messages and a 'wizard' approach to sending mail are not strictly accessibility features yet they show the importance of obtaining the input of users with a disability at the design stage. It could be anticipated that many people 'without' disabilities would also find some of these features useful. For example, those new to email may find the simple interface and 'wizard' approach of benefit.

The best way to carry out a user-centred software design involving people with disabilities warrants further research. In this paper we have presented one approach that, on the basis of feedback from involved users, was successful. The focus groups allowed group brainstorming and problem solving to occur. An interesting aspect of the group dynamic was that individuals initially tried to dominate discussion with their particular accessibility issues. However, once the participants were assured that we were aiming to achieve a universal design and a product that was inclusive of a wide variety of needs, then a group-oriented problem solving approach developed.

Running the focus groups was not without difficulty. The first consideration was ensuring accessibility for participants including accessible venues, provision of interpreters and carers and provision of accessible materials such as large print handouts for example.

Many focus group participants were inexperienced Internet users which made it difficult for them to envisage how they would use the intended email product. Presenting and discussing prototype screen shots with users was helpful with respect to this problem. There is, however, a danger with prototypes that they will unduly influence the emergence of participant's ideas. Pre-focus group education could be tried as an alternative (Sinks and King 1998).

The very nature of an individual's disability also hampered the sharing of information in some instances. For example, acquired brain injury made it difficult for one participant to follow the discussion because of short-term memory problems. Some participants with little or no speech had difficulty producing lengthy responses. Communication via alphabet boards was slow and tiring, fatigue was a significant issue for some. Also, potentially, participants with a cognitive impairment may not be able to follow the issues in a focus group forum. During focus groups, the project team and personal carers assisted individuals by going from a large group discussion to one-on-one discussion about points individuals wanted to make. These comments were written down and combined with audio-taped comments.

Given these real challenges it is important that a best practice method of eliciting requirements and design ideas is developed if the ultimate aim is to have groups of diverse users who can provide useful input to development of universally accessible software. Further research needs to be undertaken to compare a focus group approach with other potential approaches such as individual interviews, questionnaires, single disability focus groups and electronic work group approaches. An electronic group work approach, with the right software and training, may help to overcome some of the issues experienced with the focus groups while retaining the advantages inherent in a group approach. For example, it would help to overcome the issue of fatigue because participants would not be contributing in real time. Disability related communication difficulties could also be largely overcome. Such an approach would also help to make the process more cost effective as participants could be drawn from a large geographic area without the cost and effort involved in travelling.

Although considerable advances have been made in the areas of assistive technologies, accessibility features provided by operating systems and the important area of Web accessibility there is still a significant lack of tools and techniques to facilitate the easy and cost effective design and development of universally accessible applications.

Ensuring that information systems are accessible and usable for a range of users, including those with disabilities, is not an easy task and requires co-operation between a number of stakeholders. Operating system providers, application software developers, hardware developers, assistive technology providers and Web site developers all have a responsibility to contribute to the pursuit of accessible information systems (Vanderheiden 1998). However, if universally useable information systems are to be developed, users, including those with disabilities, need to be included in the design as well as the evaluation process.

REFERENCES

- Achieving Universal Access, (2000) <http://www.number-10.gov.uk/default.asp?PageID=1203>
- Bergman, E. and Johnson, E. (1997) Towards Accessible Human-Computer Interaction, *Excerpt from Advances in Human-Computer Interaction Jakob Nielsen, Editor., Vol 5.*
-

- Braithwaite, R. and Baxter, C. (1997) In *CAUSE in Australasia '97: Information Technology - The Enabler* University of Melbourne, Melbourne, Australia, pp. 24-29.
- EC (1997) *Communication on the Social and Labour Market Dimension of the Information Society. People First - The Next Steps*, European Commission.
- NOIE (2000) The Current State of Play, URL http://www.noie.gov.au/projects/information_economy/ecommerce_analysis/ie_stats/StateofPlayNov2000/index.htm
- EU (2000). eEurope 2002: An Information Society For All, (2000). European Union URL http://www.europa.eu.int/comm/information_society/eeurope/actionplan/index_en.htm
- Introduction to ISO 13407 (1999) URL <http://www.emmus.org/html/frames/guidelines/EmmusWP3/methods/iso.html>.
- Keller, S., Owens, J. and Parker, C. (2000) In *8th European Conference on Information Systems* Vienna, Austria.
- Macaulay, L. A. (1996) *Requirements Engineering*, Springer-Verlag, London.
- Owens, J. and Keller, S. (2000) In *9th Australasian Conference on Information Systems (ACIS2000)* Brisbane, Australia.
- Preece, J. (1999) *Human-computer interaction.*, Addison-Wesley, Sydney.
- QC, T. H. W. C. (2000) Maguire v SOCOG2. URL, http://www.humanrights.gov.au/disability_rights/Maguire_v_SOCOG2.htm
- Roulstone, A. (1998) *Enabling technology: disabled people, work and new technology*, Open University, Buckingham.
- Sinks, S. and King, J. (1998, March 3) Adults with disabilities: perceived barriers that prevent internet access., URL: http://www.dinf.org/csun_98/csun98_152.htm
- ABS (2000a) Statistics, Australian Bureau of Statistics. Commonwealth of Australia, Canberra.
- ABS (2000b) Statistics, Australian Bureau of Statistics Commonwealth of Australia, Canberra.
- Stephanidis, C. (1999) ERCIM ICST Research Report.
- Stephanidis, C. and Emiliani, P. L. (1999) 'Connecting' to the information society: a European perspective, *Technology and Disability*, 10, 21-44.
- Story, M. (1998) Maximizing Usability: The Principles of Universal Design., *Assistive Technology*, 10,1, 4-12.
- Sun Microsystems' Enabling Technologies Program, www.sun.com/tech/access
- Universal Design Research Project, (1998) http://trace.wisc.edu/docs/univ_design_res_proj/udrp.htm
- Vanderheiden, G. (1992) Trace Research and Development Center, Madison, Wisconsin.
- Vanderheiden, G. (1994, June 1) Application Software Design Guidelines: Increasing the Accessibility of Application Software to People with Disabilities and Older Users., http://trace.wisc.edu/docs/software_guidelines/toc.htm
- Vanderheiden, G. (1998) Cross-modal access to current and next-generation Internet - fundamental and advanced topics in Internet accessibility, *Technology and Disability*, 8, 115-126.
- Vanderheiden, G. (2000) In *Conference on Universal Usability* Washington, DC, United States, pp. 32-38.
- Vanderheiden, G. C. and Vanderheiden, K. R. (1992) Trace Center.
- Vocational Rehabilitation Branch - ILO (1999). *Proceedings of the International Symposium on job retention and return to work strategies for disabled workers*. Geneva: International Labour Office.
- WC3 (1999) Web Content Accessibility Guidelines 1.0., URL <http://www.w3.org/TR/WAI-WEBCONTENT/>

ACKNOWLEDGEMENTS

The MultiMail research and development project was funded by the Commonwealth Department of Communication, Information Technology and the Arts' AccessAbility Program. The MultiWeb research and

development project was funded by the Commonwealth Department of Health and Family Services. We would also like to acknowledge the contribution of Paul Spain of the Excellent Programming Company (paul.spain@bigfoot.com) for his work on the design and development of MultiMail in particular the word prediction software and email back-end.

COPYRIGHT

Susan Keller, Richard Braithwaite, Janet Owens and Kaye Smith © 2001. The authors assign to ACIS and educational and non-profit institutions a non-exclusive licence to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive licence to ACIS to publish this document in full in the Conference Papers and Proceedings. Those documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors.