Technical Report TR 2-2005-0708

Lessons learned from Mobile Application Design for Health Care

Andreas Holzinger, Maximilian Errath

Institute of Medical Informatics, Statistics and Documentation (IMI)

Graz Medical University, Auenbruggerplatz 2

A-8036 Graz Austria

andreas.holzinger@meduni-graz.at

maximilian.errath@meduni-graz.at

Abstract:

Designing Web-applications is considerably different for mobile computers (handhelds, Personal Digital Assistants) than for desktop computers. Screen size as well as system resources are limited and end-users interact differently. Consequently, detecting handheld-browsers on the server side and delivering pages optimized for a small client form factor is inevitable. The authors discuss their experiences during the design and development of an application for medical research, which was designed for both mobile and personal desktop computers. The investigations presented in this paper highlight some ways in which Web content can be adapted to make it more accessible to mobile computing users. As a result the authors summarize their experiences in design guidelines and provide an overview of which factors have to be taken into consideration when designing software for mobile computers.

Keywords: Information Interfaces and Representation, Interface Design, Mobile Computing, Life and Medical Sciences, Internet Applications

"The old computing is about what computers can do,
the new computing is about what people can do" [54].

1. Introduction

1.1. Mobile Computing as a challenge for Universal Access in the Information Society

Mobile Computing is an umbrella term and describes any technology that enables *people* to access information and supports them in daily workflows independent of location. Actually, it is remarkable that many of the Mobile Computing research results and breakthroughs came from the Human-Computer Interaction (HCI) field. However, there is still room for a great deal of progress in this extremely successful and worthwhile area of endeavor. The phenomenal growth of Mobile Computing, which has not been accompanied by an equivalent increase in the understanding of end-users for the complexities of this subject, has caused an increase in the need for research in mobile computing and a parallel campaign to increase the awareness and proficiency of the end-users.

Consequently, the area of mobile computing is a good example of the *new computing* in the sense of Ben Shneiderman: the New Computing technologies must enable users to accomplish their tasks and to relax, enjoy, and explore [54].

1.2. Why Mobile Computing in Health Care and Medicine?

A representative example of mobile computers is the handheld computer, which is often referred to as a Personal Digital Assistant (PDA). This device presents a number of challenges in Human-Computer Interaction (HCI) and Usability Engineering (UE) including the tension between the appropriate user interface design, the device and the social context of the device's use [42].

Handheld computers are often used together with desktop computers [43] and subsequently support nomadic and ubiquitous computing [30]. This is of tremendous interest for physicians and healthcare professionals.

Medical doctors and nurses work in an environment which requires high mobility. Within their daily routine their sphere of activity alters frequently between wards, outpatient clinics, diagnostic and therapeutic departments and operating theatres. Although access to stationary clinical workstations is provided in the hospital, their locations do not always coincide with the user's current workplace. In order to fulfill a high health service standard, the medical staff has an extensive demand for information at a number of locations – which actually only mobile computers can supply [52]. For example: Up-to-the-minute electronic patient record information is not always available at the bedside [6], [68]. New orders or diagnostic results noted during rounds must be transcribed to the electronic patient records via a clinical workstation at a later time – whereas a mobile computer enables direct access [41], [31].

However, although mobile computers have been available for a relatively long time [9] in hospitals, different studies [4] show that health care professionals are reluctant to use poorly designed mobile systems, as the work at the point of patient care is very time-pressured and hectic.

To design and develop mobile systems with high acceptance, it is essential to obtain empirical insight into the work practice and context in which the mobile system will be used. Consequently, mobile devices are only useful when design and software validation aspects have been taken into account [24].

1.3. Usability and Usefulness

As it is difficult to make devices with small displays **usable,** there is also the fundamental challenge of **making them useful.** With useful we mean that the application is beneficial and can help the end-users to achieve a particular goal, whereas with good usability we mean that the application is easy and pleasant to use.

To achieve *both* goals, particularly for mobile devices, it is strongly recommended to apply a User-Centered Design (UCD) approach [36]. UCD evolved in the field of HCI and was first articulated as such in User Centered System Design [47] and focuses strongly on the intended end-users [62].

1.4. Objectives and structure of this paper

Applications for mobile computers are often the *mobile part* of a larger non-mobile Web application. In this case the mobile solution must be particularly designed for a quick and short interaction *on route*. This was of particular importance for our highly mobile group of end-users, i.e. medical doctors and healthcare professionals within a Hospital. We intended to raise the acceptance of the main system by the availability of a mobile solution – in the sense of ubiquitous usability.

In this paper, we aim to highlight some issues in which Web content can be adapted to make it more accessible to mobile handheld computer users. The overall aim of our project was to provide automatic adaptations of content; enabling users to gain access to as wide a range of the material as possible.

In chapter 2, we provide some details about the problem to be solved and the resulting application. In chapter 3, we describe our design and development issues and in chapter 4, we present and discuss some of the issues we derived, in the form of condensed guidelines.

2. The Randomizer for Clinical Trials

2.1. Randomization in Clinical Trials

In medical research, the randomized clinical trial is the gold standard for assessing treatment effects [8]. Randomization - in this context - means, for example, in a trial of a standard treatment versus a new treatment, random allocation of patients to treatment groups ensures that each patient has the same chance of receiving either the new or the standard treatment. There are two main reasons why randomization is used. First, randomization leads to treatment groups that are random samples of the population and thus statistical methods based on probability theory can be used. Second, it minimizes bias thus leading to groups that are generally comparable, thereby ensuring that observed differences between the treatment groups are due to differences in the treatments alone. Random allocation does not guarantee that the groups will be identical apart from the treatment given but it ensures that differences between them are due to chance alone. For more information on randomization and good practices in clinical trials see, for example, the ICH (www.ich.org) Guidelines or the CONSORT (http://consort-statement.org) statement.

In most clinical trials randomization is performed using pre-printed randomization lists, sealed envelopes, telephone-based services (for example using automated response technology), or in large and multi-center clinical trials (i.e. trials where multiple centers are contributing patients) a commercial third party - the trial coordinating center - is responsible for patient registration, data collection and randomization.

2.2. Web-based solution

Performing randomization and trial data collection via the Internet using Web-based applications has several advantages over the traditional telephone-based services, particularly in multi-center trials.

Using the web reduces communication delays and expenses, provides a worldwide 24-hour-service, reduces transcription errors, supports better auditing due to comprehensive logging of each transaction, and saves the researchers time. Finally, yet most importantly, a web-based solution also facilitates communication between the trial users (investigators, statisticians, trial coordinators, etc.).
For example, the latest version of the trial handbook or a directory of trial users' email addresses can be stored on the central website used for randomization and downloaded by trial users on demand.

2.3. From Web-based to Web-based mobile solution

A web-based solution for randomization in multi-center clinical trials was implemented, which has been called the **Randomizer.** The desktop version of this application was that successful that it is meanwhile commercially available, see: www.randomizer.at

The Randomizer provides a self-serve, easy to use, secure and 24 hour-a-day randomization service that runs exclusively on the Internet. Investigators randomize patients with only a few clicks by completing an on-screen form with patient details and are immediately notified of the treatment allocation. Using a flexible role-based access control, trial coordinators can nominate investigators, biometricians, monitors and pharmacists. For multi-center trials, user management can be delegated to participating centers. Trial coordinators can open or stop patient recruitment at any time. They can also choose to be notified via email on events such as randomization, emergency un-blinding and others. Many methods of dynamic randomization, such as permuted blocks, biased coin, urn randomization, minimization and others, are available. As an additional service for biometricians, the Randomizer provides a powerful simulation tool, which can be used to generate static randomization lists, to test a trial design or to study the implemented randomization methods. All transactions are logged. The trial's audit trail and the list of randomizations can be downloaded and analyzed at any time by the trial monitor. Network traffic between the web-browser and the Randomizer is encrypted using SSL (Secure Sockets Layer) with strong (128 bit) encryption. SSL is the industry standard security technology for creating an encrypted link between a web server and a browser and is widely used for Internet banking and e-commerce.

To make the core functionality of this software – i.e. randomization of new patients – available in places without an infrastructure to support computers and networks, we considered using handhelds equipped with cell phone (GSM) or WLAN modules. This offers a rapid solution for physicians who work with existing healthcare practice habits and flows with the healthcare professionals' daily routine (see figure 1).

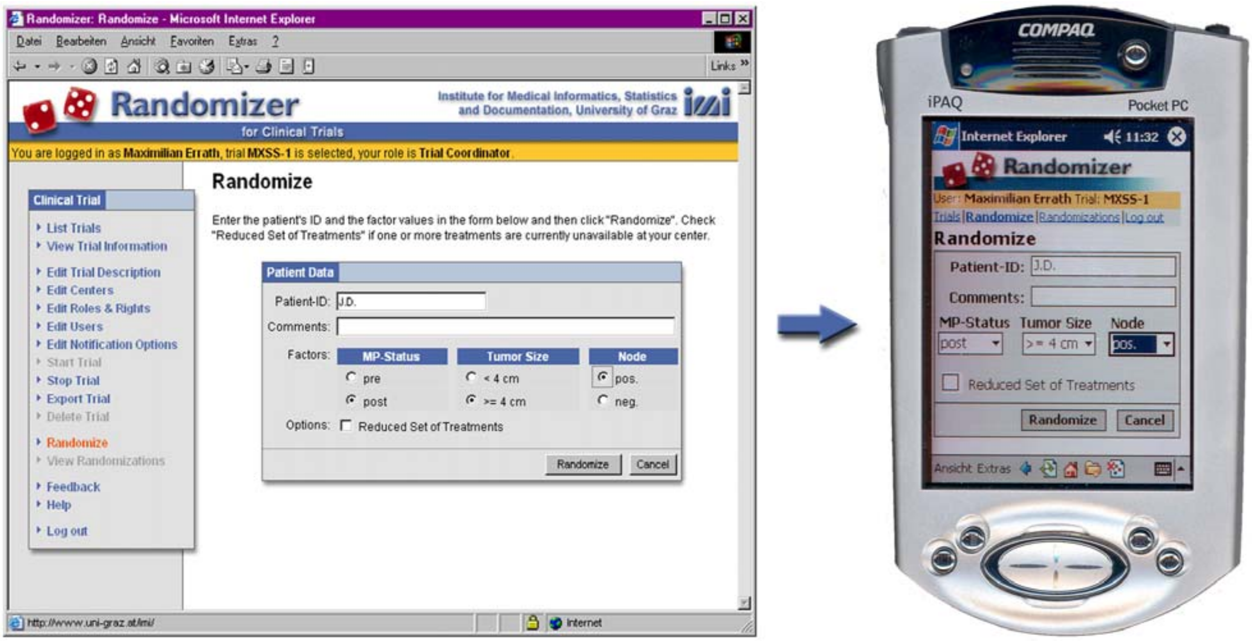

Fig. 1 The randomization software offers patient randomization and trial management features using a standard web-interface designed for desktops. Core functionality (i.e. randomization) is also available via web pages optimized for PDAs.

3. Design, Development, Experiments & Evaluation

3.1. User Centered Development

During the development of the Randomizer we applied the *User Centered Development* method. Although User Interface Design for mobile computers and usability evaluation of software for mobile computers has been an emerging area of research for some time, most available work has reported about techniques for evaluating the usability of mobile computers within laboratory settings [29], but interaction of end-users must be studied in a **real life setting** [19], [20].

It is also generally known that software developers rarely use Usability Engineering Methods (UEM) on software development projects [22]. This even includes such basic usability engineering techniques as early focus on the end-user, empirical measurement and iterative development of prototypes [45]. Few projects to date have adopted a fully integrated User Centered Development approach in one strategic shift [19], [20].

Important is an expansion from User Centered Design to User Centered Development [35], [21]. Every user centered approach involves knowing who the end-users will be [32], [1] including their capabilities and limits, needs and expectations, their goals and the tasks required to achieve those goals, as well as the physical and social environments and the context in which end-users must achieve those goals [55]. One of the most critical aspects is to hold the motivation of a user at a satisfactory level [16]. This involves processes of participatory design, end-user testing and iterative design [13], [53]. Also with consideration of the International Standard 13407 [65], prototyping is a vital element during development [21], [23].

3.2. Prototyping

Whilst designing an Interface, designers are instantly confronted with a dilemma: principally it is not possible to evaluate an interface until it is built – but after it has been built, changes are difficult and expensive. One solution to surmounting this dilemma is prototyping. This means producing cost effective interactive versions of an interface design. Together with Usability Engineering Methods, this enables us to bring the end-user into the development process at a very early stage. Some advantages of prototyping include: Reduction of development time and consequently a reduction of costs; the requirement of end-user involvement from the beginning, thus best suitable for user-centered development; Facilitation of system implementation, since the end-users know what to expect, consequently resulting in higher end-user satisfaction. Some disadvantages of prototyping include: Might possibly lead to insufficient analysis, resulting in incomplete documentation; the end-users might expect the performance of the final system to be "identical" to the prototype; Developers can become too attached to their prototypes.

Working prototypes vary according to the breadth or depth of features implemented. Working prototypes cut down on either the number of features, or the depth of functionality of features: *Vertical Prototype:* in-depth functionality for a few selected features; *Horizontal Prototype:* full interface features, but no underlying functionality; *Scenario Prototype:* only a functionality of specific scenarios or paths through the interface;

However, prototyping is always a quick way to incorporate direct feedback from end-users into a design. Paper based prototyping bypasses the time and effort required to create a working, coded user interface. Instead, it relies on simple tools including paper, scissors, stickers etc. [51], [21].

3.3. Paper Mockups

Paper mockups do not need to incorporate all the frills of technology, they only need to capture the site's functionality and convey the right information [15].

The term *paper mock-up* means "to prototype the screen designs and dialogue elements on paper". It is certainly the easiest and most efficient method. With common office supplies, each interface element (e.g. dialog boxes, menus, error messages etc.) can be sketched and hand-printed on pieces of paper and index cards.

In fact, they *cost little* to produce and *encourage more suggestions* since they are obviously *easily changed.* This leads to an *easy creation of alternatives.*

Seeing as every first prototype interface has flaws, with a paper mock-up it's easy to just scribble out a changed element and see if that fixes the problem. Our experiences showed that people were more likely to change the paper mock-up than (later) the high-fidelity prototype. This is also evident from previous knowledge: people usually take paper prototypes seriously; they find many usability problems [21], [44], [33].

Some advantages of Paper mock-ups include:

- First sketches allow early and immediate usability feedback;
- Because it is not too detailed, designers and end-users can concentrate on abstract dialog concepts and not on technological details;
- Cheap to produce (sketches, easy to use, fast turn-around, can encourage team design, early usability feedback with throwaway designs results in a maximum feedback for minimum effort);

Some disadvantages of paper mock-ups include:

- It is relatively difficult to capture interface behavior (see e.g. the Wizard of Oz Technique, [37]);
- In combination with usability methods (see chapter 3.4) it needs more time than theoretically predicted (see e.g. [21]);
- There is still low confidence in this method amongst software developers, due to a failure to make them aware of these methods during their engineering education [20].

Fig. 2 Paper Mock-ups proved to be efficient within the development of the Randomizer

3.4. Experimental design and Methods

During the design of the desktop version of the Randomizer, especially during the analysis of the workflows of the medical personnel, soon the need for a mobile application emerged. However, since the requirements to a mobile solution are definitely different, we developed a three-level plan in order to both follow a end-user centered development and to generally gain insight into the challenges of mobile applications:

Level 1) Investigations on how the end-users would be best supported by a mobile application;
Level 2) User-Centered Development on the basis of paper-mock ups;
Level 3) Experiments, in order to gain insights into the differences between desktop and mobile application.

Ad level 1) At first, it was very important to carefully understand and specify the *context of use*, the characteristics of the end-users, as well as the organizational and physical environment where the end-users usually will use their application. Due to the fact that the end-users' tasks characterize the context in which the mobile solution is used, we defined a task as the way in which a specific goal is attained. Consequently, we carefully analyzed the following issues:

- Characteristics of the end-users (this includes knowledge, skill, experience, education, training, physical attributes, habits and capabilities).
- Specific tasks that the end-users are to perform (typical scenarios, frequency and duration of performance etc.).
- Environment in which the end-users will use the mobile application (this included network infrastructure, workplace, work practices and attitudes).

Ad level 2) In order to gain insight into the issues as described above and to carry on with our development we carried out *thinking aloud studies in real-life settings.* For the thinking aloud studies with paper-mock ups we used the following equipment:

- Hi8 video camera and recorder on tripod;
- mirror;
- high qualtiy microphone (fixed on tripod, acoustically decoupled from camera to avoid interferences);
- headphones (monitoring the sound is essential);

Parallel, we checked several of these issues with questionnaires and by orally interviewing the end-users (see section 4 for some of our experiences and lessons learned).
On the basis of the gained insights we implemented the prototype in a user-centered design approach.

Ad Level 3) Finally, we tested specific tasks involving careful selected N = 12 end-users (see chapter 3.5) by comparing the task performance between the desktop application and the mobile application (see chapter 3.6). For these experiments in real-life we used a stopwatch.

3.5. Participants

Previous experiments [66], [46] showed that more than 80 % of all usability problems can be found by examining around 3 to 5 end-users.

Consequently we involved N = 12 end-users consisting of 4 Novices, 4 Intermediates and 4 Experts or Semi-experts respectively. We did not test in a usability lab but in a real-life setting, which is not easy with mobile devices, but we found to be absolutely necessary to fully understand the end-users' context as described in chapter 3.4.

3.6. Task Performance Time

During our task experiments we concentrated especially on:

- number and type of errors per task;
- number of errors per unit of time;

- number of navigations necessary;

and most of all on

- time to complete a task (time to perform task, ttpt).

The *time required to perform a specific task* is still the most important factor to measure the value of an application [60], [59], [61]. A specific task on handheld devices requires more *time to complete task* than the same task on desktop computers (figure 3.)

For the experiments we used Hewlett Packard iPAQs, which comes with a 240 x 320 pixel display; however, with Microsoft's Pocket Internet Explorer (Pocket IE) the browser decoration (scroll bars, address line, title bar, etc.) further limits the visible content area to 229 x 255 pixels.

There was a strong correlation between **time** to complete the task and **problems** in locating functions and navigation. Subsequently, pages that need horizontal scrolling should be avoided. Most of the end-users overlook the fact that there is more to view in horizontal direction. Scrolling can be most simply reduced by strictly focusing on the content task.

Due to the limited functionality of the PDA application – in contrast to the Randomizer's native web interface – navigation paths are short and limited to two levels in depth. Therefore navigation should not be an issue for the increased *time to complete task* observed during our experiments.

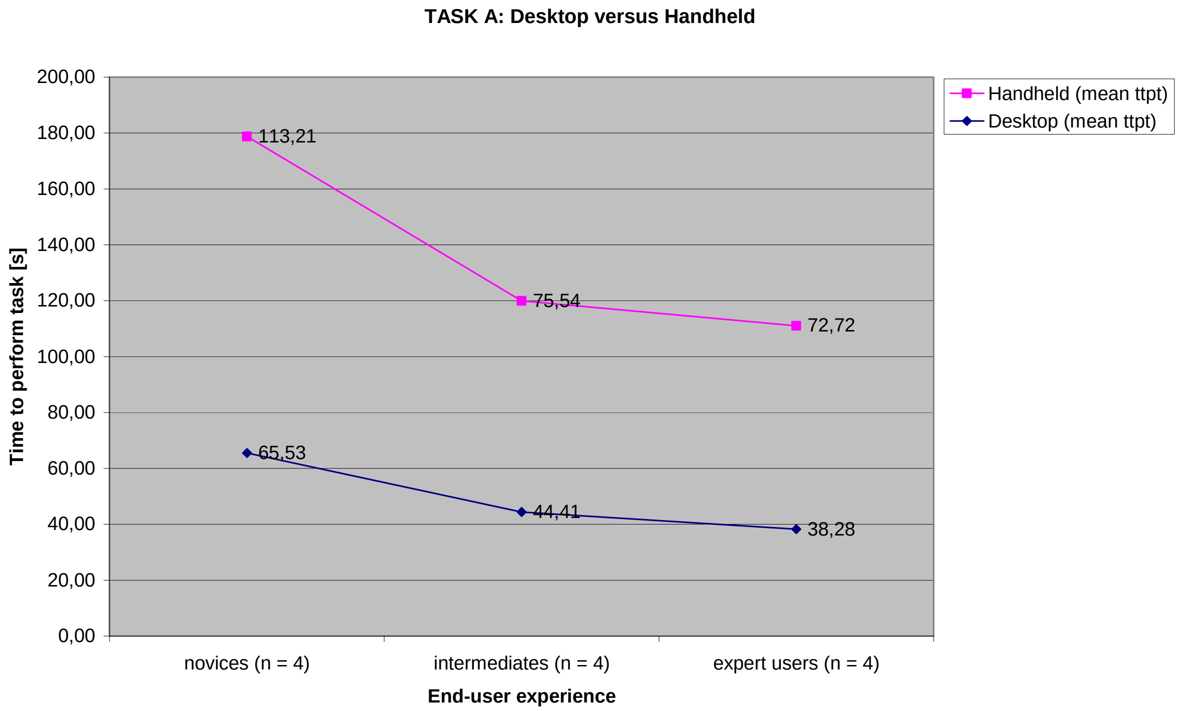

Fig. 3: *Time to Perform Task:* Desktop versus Handheld

4. Lessons Learned in Developing Web-Applications for Handhelds

Based on our experiments (refer to chapter 3.4), we saw that developing Web-applications for handhelds is considerably different than for desktop computers [27], [18]. Different factors are to be considered within the following problem areas:

- Information output (chunking, textual information, graphics, Icons, colors, errors, help);
- Information input (tapping, circling, typing);
- Navigation (interaction style, menus, scrolling, browsing, shortcuts); and
- Connectivity (handling connections, synchronizing information, security) – factors which will not be discussed within this paper.

We present our experiences during the development of the Randomizer in the form of condensed guidelines. Although general guidelines are available (see e.g. the NIH Web design guidelines, available online at http://usability.gov/guidelines/guidelines_notice.html) none are known to us, which specifically address mobile computing in medicine.

4.1. Information output on small screens

4.1.1. Chunking information

Humans are able to expand the capacity of their working memory by *chunking information* together [39]. A chunk is defined ambiguously and subjectively depending on the previous knowledge of the end-users [58].
For example: four letters can be four (separate) chunks (PXHF) or just one chunk (ICON). Subsequently, a combination of letters which form a pronounceable syllable will always be more easily remembered than one that is not pronounceable.

Based on our research we propose the following guidelines:

- Display the most relevant information first, and allow the users to choose which information they want to view next;
- Highlight relationships between information: proper chunking of information is essential on small screens;
- Present the information in a way which is familiar to the end-users – know your end-users, e.g. present information in the way the end-users need it; base your design on particular experiments within the specific context!
- Locate common information in common screen locations;
- Provide the frequently used information in a location most obvious to the end-users;
- Place general items in precedence of specific items; if no order exists: alphabetize.

4.1.2. Textual Information

The relationship of screen size and textual information on the ability to understand this textual information was already examined before the era of the Web [40]. It is interesting to note that research on screen reading showed that the reading speed decreased approximately 30 % whilst reading on large screens as compared to reading on paper [40].

As computers became more and more widespread, the readability on large screens increased but is still less than reading from paper [69]. The evolution in readability on small screens is not likely to follow the same pattern; decreased readability will still be intrinsic to limited screen size [69].

Duchnicky & Kolers (1983) [7] considered the effect of window height and line widths on reading: The full width display was read 25% faster than the screen which was 1/3 the width. The impact of varying the display height was very much less remarkable.

According to our experiences the following guidelines should be taken into consideration:

- Keep the amount of text within the application generally to a minimum;
- Consider that the end-users typically make use of the mobile application away from their office, consequently text should be designed for conditions where reading text is difficult;
- Use sans serif fonts (Tahoma, Verdana) and avoid small fonts, be consistent in the use of font, font size, styles etc.;
- Present the most important content first (Inverted Pyramid Style);
- Do not right justify text (this causes interruption of eye movement and an obstruction to reading);
- Provide good contrast of the text against the background (black fonts on white background is simple and is most often the best; see section color for more details);
- Present text in small interfaces carefully: avoidance of abbreviations and truncations;
- Design interfaces that display long text paragraphs in a way, which is more suitable for small screens, e.g. by using adaptive Rapid Serial Visual Presentation (RSVP), which

adapts the presentation speed to the characteristics of the text instead of keeping it fixed [48].

4.1.3. Icons

Icons are pictorial representations and suggest a certain meaning. There are many different guidelines for building icon-based interfaces (examples include [2], [38], [25]). Icon design can be limited by implementation constraints and are sometimes restricted by those that are in existence already, thereby lacking originality in some sense [10].

Originally, Icon design evolved from the concept of signs according to Charles S. Peirce [50], [18], [17]. Peirce viewed a sign (S) as a product of a three-way interaction between a so called *representamen* (that which represents), the sign's object (that which is represented) and its mental so called *interpretant* (the process of interpretation): S = S (R, O, I)

This implies that for User Interface Design icons alone are meaningless without a particular *context* and in Human–Computer Interaction the following relationship for icons results [12], [38]: Icon + context + viewer > meaning.

The following design guidelines result from these theoretical considerations in combination with our practical experiences:

- Replace textual information through representative and familiar Icons whenever possible;
- Provide familiar Icons only, do not force target end-users to learn new icons, they must always be familiar; if simplicity cannot be provided – use text instead of a new Icon;
- Be aware that the meaning of Icons is culture dependent, consequently test icons with target end-users;
- Provide good contrast between the Icon and the background and ensure that Icons are distinguishable from each other; shape, form and structure of an Icon must always be unique;

4.1.4. Colors

Very important is the proper use of color [56], but most of the research on color predates the Web [49], [34]. It is very interesting that 100 years after Isaac Newton, Johann Wolfgang Goethe (1749-1832) examined the problems of color and although his *Theory of Colors* intended to attain *"a more complete unity of physical knowledge"* by including all branches of the natural sciences, Goethe approached the subject primarily to gain some knowledge of colors *"from the point of view of art"* [11]. We cannot emphasize enough that color is one of the most efficient techniques for transferring visual information from computer to human and the impact of color, as well as the problems of color perception, has been described for a long time [14]. There are general guidelines available, e.g. [57], [67], but unfortunately most guidelines that currently exist are often not based on experimental research [64].

Based on the results of previous experiments on the proper use of color, we can recommend:

- Use colors to help the end-users to orientate and navigate; but never use a color without special intention;
- Use a consistent color scheme within the whole application; use colors to group information;
- Make sure that all information conveyed with color is also readable without color;
- Avoid under all circumstances, the use of too many colors together; avoid whenever possible the combination of red and green;
- Avoid saturated colors (they cause visual confusion and stimulate the eye enormously);
- Be careful when using blue: thin blue lines are visible but usually difficult to perceive, black and white always provide the best resolution for fine details and small shapes;
- Be careful when using colors of similar brightness or contrast, because they might be difficult to distinguish;
- Always test the colors used with your target end-users, because colors generally have a strong culturally dependent meaning;

4.1.5. Errors

Humans should always be expected to err, consequently applications must be designed to tolerate and prevent user errors. This is especially true for mobile devices because we experienced that end-users are far less familiar with mobile devices than with standard devices (see also figure 3), therefore **always:**

- provide an easy “back” button – every end-user action must be reversible;
- accept common misspellings;
- require a confirmation if the requested action causes a change;
- allow expert users to disable confirmation requests by the application settings;
- provide meaningful error messages in plain language;

4.1.6. Help

Although many of the end-users feel comfortable having help functions available on request, we were of the opinion that our application must be useable *without any help function,* consequently we did not implement any online help.

4.2. Information Input

Entering information into handhelds is possible without typing text. There are also other possibilities including voice recognition – which can be interesting in clinical applications – or separate enhancements (camera, keyboards, keypads etc.).

However, textual information is still the most important. The common methods are typing via virtual keyboards and selection methods including tapping and circling.

- Prefer general selection controls instead of text entry (the error rate resulting from selecting is generally lower than the error rate resulting from entering data; again consider that the users might use the handheld away from their offices;
- Thoroughly understand the users' task, in order to avoid long or unnecessary text entry or information input at all; text entries seemingly unnecessary to the end-users is easily left unused;
- Allow the application to "learn" from the users input, consequently you can enhance input of repeatedly used text entry;
- Prioritize information entry especially when converting a desktop application to a handheld application;
- Support copy-and-paste functionality (generally known as *cut and paste*);
- Allow end-users to increase/decrease numbers by using the navigation key;
- Provide default values whenever possible; use previous entered data as default value; again: a thorough task analysis is necessary in order to know exactly how the end-users work;
- Provide default values or at least zeroes for numerical data to indicate the data format;
- Enable the users to exit the application quickly, without losing any information; however, the application should remember what the users have done previously and start at this exit point again (when applicable);

4.3. Interaction and Navigation

Good Navigation is the primary key to good usability. Orientation and Navigation are considered to be essential for usability issues [3]. Navigational actions compose nearly 90% of *all* Browser actions [63]. Consequently, to use any Web-based Information System as effectively as possible, the quality of the navigation support is of primary importance.

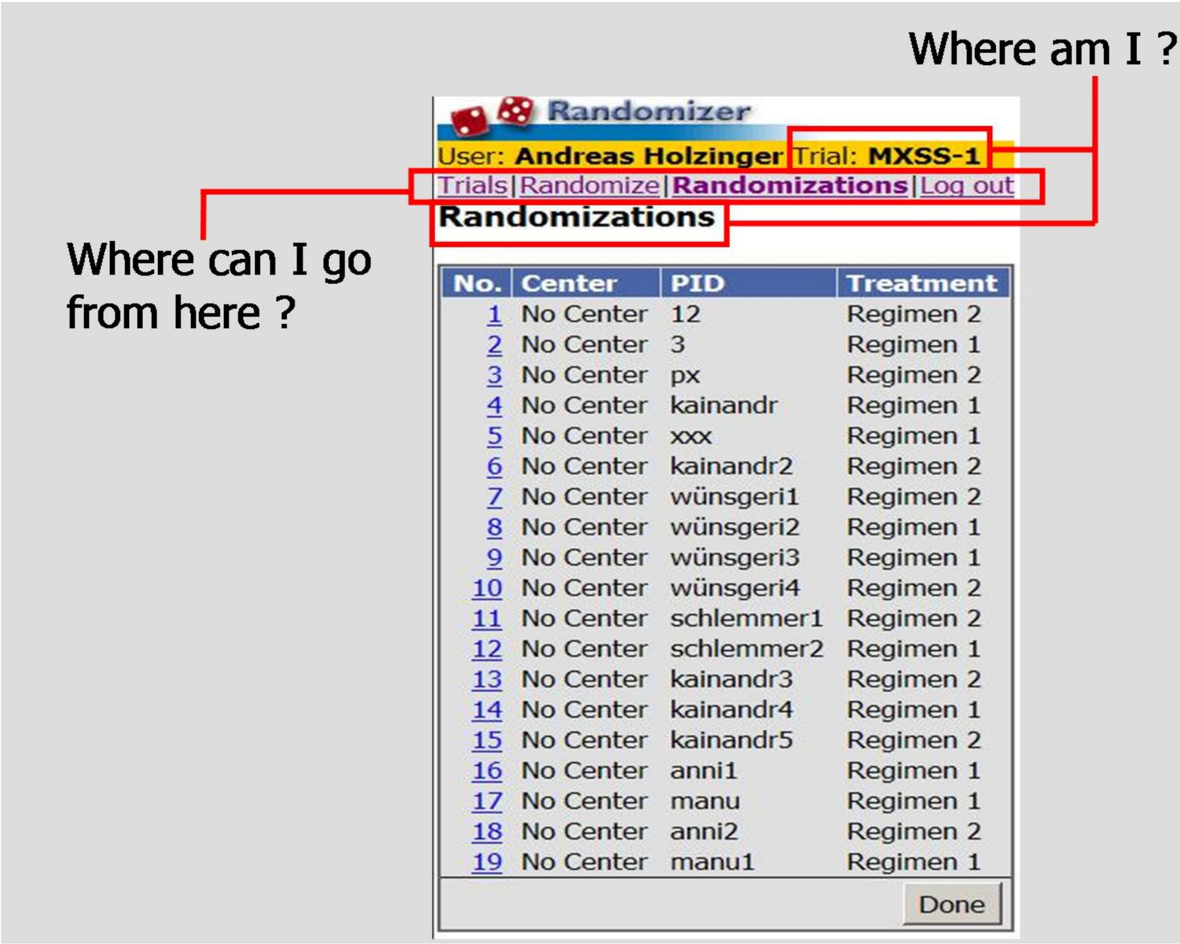

Fig. 4 : The most important questions to be answered during our experiments included: "Where am I?" and "Where can I go from here?"

Problem areas concerning interaction and navigation can be subdivided into interaction style, scrolling and browsing.

End-users **interact differently** with handheld devices than with PCs (see for example [36], [26], [5]). For example, where it is acceptable to require keyboard input on desktop machines, typing on a PDA's soft-keyboard becomes getting burdensome. With handheld devices it is much easier to select values from a list rather than enter text in an input field. Concerning scrolling: Due to the increased web experience of the end-users, is not as bad as in previous years [28].

Summary Guidelines on Interaction and Navigation:

- Provide the core features of the applications from the main view of the applications;
- Ensure that the navigation keys never result in an unexpected action that the user cannot anticipate;
- Provide a context-sensitive menu if there is no single intuitive default action;
- Use softkeys consistently;
- Use all terms consistently;
- Avoid scrolling;

5. Conclusion

The availability of a mobile solution raised the acceptance for the main system. Our investigations highlighted some ways in which Web content can be adapted to make it more accessible to users of handheld mobile computers. It is essential to provide automatic adaptations of content, for example by automatic preprocessing or providing different versions based on screen resolution and browser capabilities, hence end-users can gain access to as wide a range of the material as possible.

However, the guidelines we have described here, are applicable to content which is to be specifically designed for small display platforms. The screen size is limited, for example a typical handheld, including the Hewlett Packard iPAQ, comes with a 240 x 320 pixel display. With Microsoft's Pocket Internet Explorer (Pocket IE) the browser decoration (scroll bars, address line, title bar, etc.) further limits the visible content area to 229 x 255 pixels. A first experiment – using the standard web-interface designed for desktop machines – did not lead to useful results. Users simply got lost whilst scrolling horizontally and vertically through the pages. Pocket IE's *Fit to Screen* feature *did not help* since our page layout was too complex for reformatting to such a small screen resolution. Moreover, our page design depends on cascading stylesheets (CSS) and Pocket IE available with the version of our Pocket PC operating system has *no* CSS-support built in (XSL stylesheets can also be used). Therefore, detecting PDAbrowsers on the server side and delivering pages optimized for a small client form factor is inevitable. The most important guidelines include: Keep things as simple as possible. Every mobile device has limited resources.

We recommend a simple, mainly text-based interface with few small images. Pages must always be designed to allow dynamic resizing, fixed-size designs (e.g. using tables and transparent images for sizing table columns), and pages that need horizontal scrolling must be avoided.

We realized that most of our end-users overlooked information, which required horizontal scrolling. Scrolling can most simply be reduced by strictly focusing on the content task. Mainly *flat menu hierarchies* and simple *single column* layouts proved to be appropriate. Quick and easy navigation for frequent functionalities are an absolute necessity. Our experimental results emphasized again that extensive text entry is to be avoided and that the priority of shortlists over direct input is important. User Centered Design (UCD) proved to be an appropriate design technique for this kind of mobile computer interface design and finally led to a simple and easy to use interface, in accordance with the proverb: *less is more.*

6. References

1. Andrews K (2001) Web Usability on the Cheap. In: Holzinger A (ed) Human-Computer Interaction in the 21st Century, Austrian Computer Society, Vienna, pp. 83-95
2. Benbasat I, Todd P (1993) An experimental investigation of interface design alternatives: icon versus text and direct manipulation versus menus. International Journal of Man-Machine Studies, 38 (3): 369-402
3. Benyon D (2001) The new HCI? navigation of information space. Knowledge-Based Systems, 14 (8): 425-430
4. Brekka T (1995) Select mobile computers tailored to healthcare environment. Health Management Technology, 16 (13): 48, 50
5. Brewster S (2002) Overcoming the Lack of Screen Space on Mobile Computers. Personal and Ubiquitous Computing, 6 (3): 188-205
6. Choi J, Chun J, Lee K, Lee S, Shin D, Hyun S, Kim D, Kim D (2004) MobileNurse: hand-held information system for point of nursing care. Computer Methods And Programs In Biomedicine, 74 (3): 245-254
7. Duchnicky RL, Kolers PA (1983) Readability of text scrolled on visual display terminals as a function of window size. Human Factors, 25 (6): 683-692
8. Field D, Elbourne D (2003) The randomized controlled trial. Current Paediatrics, 13 (1): 53-57
9. Forman GH, Zahorjan J (1994) The Challenges of Mobile Computing. IEEE Computer, 27 (4): 38-47
10. Gittins D (1986) Icon based Human-computer interaction. International Journal of Man-Machine Studies, 24: 519-543
11. Goethe JW (1810) Farbenlehre (Theory of Colors, English from Charles Eastlake, 1970). Kohlhammer, Stuttgart (Germany)
12. Goonetilleke RS, Martins Shih H, Kai On H, Fritsch J (2001) Effects of training and representational characteristics in icon design. International Journal of Human-Computer Studies, 55 (5): 741-760
13. Gould JD, Lewis C (1985) Designing for usability: key principles and what designers think. Communications of the ACM, 28 (3): 300-311
14. Hering E (1961) Principles of a New Theory of Color Sense", in COLOR VISION. In: Teevan RC, Birney RC (eds) Color Vision, van Nostrand, Princeton (NJ), pp.
15. Hix D, Hartson HR (1993) Developing user interfaces: ensuring usability through product & process. Wiley, New York
16. Holzinger A (1997) Computer-aided Mathematics Instruction with Mathematica 3.0. Mathematica in Education and Research, 6 (4): 37-40
17. Holzinger A (2002) Multimedia Basics, Volume 1: Technology. Technological Fundamentals of multimedial Information Systems. Laxmi Publications, New Delhi
18. Holzinger A (2002) Multimedia Basics, Volume 3: Design. Developmental Fundamentals of multimedial Information Systems. Laxmi Publications, New Delhi

19. Holzinger A (2002) User-Centered Interface Design for disabled and elderly people: First experiences with designing a patient communication system (PACOSY). In: Miesenberger K, Klaus J, Zagler W (eds) Lecture Notes in Computer Science. Vol 2398, Springer, Berlin et al., pp. 34-41
20. Holzinger A (2003) Experiences with User Centered Development (UCD) for the Front End of the Virtual Medical Campus Graz. In: Jacko JA, Stephanidis C (eds) Human-Computer Interaction, Theory and Practice, Lawrence Erlbaum, Mahwah (NJ), pp. 123-127
21. Holzinger A (2004) Application of Rapid Prototyping to the User Interface Development for a Virtual Medical Campus. IEEE Software, 21 (1): 92-99
22. Holzinger A (2004) Usability Engineering for Software Developers. Communications of the ACM, in print
23. Holzinger A (2004) Usability Engineering Methods. Online at: http://webdb.uni-graz.at/~holzinge/holzinger/usability.html (last access: August, 15, 2004)
24. Holzinger A, Errath M (2004) Designing Web-Applications for Mobile Computers: Experiences with Applications to Medicine. In: Stephanidis C, Stary C (eds) User-Centered Interaction Paradigms for Universal Access in the Information Society. Lecture Notes of Computer Science. Vol. 3196, Springer, Berlin, Heidelberg, New York, pp. 262-267
25. Huang S-M, Shieh K-K, Chi C-F (2002) Factors affecting the design of computer icons. International Journal of Industrial Ergonomics, 29 (4): 211-218
26. Jessup LM, Robey D (2002) Issues and challenges in mobile computing: The relevance of social issues in ubiquitous computing environments. 45 (12): 88-91
27. Karampelas P, Akoumianakis D, Stephanidis C (eds) (2003) User Interface Design for PDAs: Lessons and Experiences with the WARD-IN-HAND prototype. Springer, Berlin, Heidelberg, New York
28. Kim L, Albers MJ (2001) Web design issues when searching for information in a small screen display. In: 19th annual international conference on Computer documentation, Santa Fe (NM), pp 193-200.
29. Kjeldskov J, Stage J (2004) New techniques for usability evaluation of mobile systems. International Journal of Human-Computer Studies, 60 (5-6): 599-620
30. Kleinrock L (1996) Nomadicity anytime, anywhere in a disconnected world. Mobile Networks and Applications, 1 (4): 351-357
31. Konstantakos AK (2003) Personal computers versus patient care: at the desktop or at the bedside? Current Surgery, 60 (4): 353-355
32. Krug S (2000) Don't Make Me Think: A Common Sense Approach to Web Usability. New Riders, Indianapolis (IN)
33. Lewis JR (1994) Sample sizes for usability studies: additional considerations. Human Factors, 36 (2): 368-378
34. Marcus A (1986) The Ten Commandments of Color. Computer Graphics Today, 3 (11): 7,12,14
35. Marcus A (2002) Dare we define user-interface design? interactions, 9 (5): 19-24
36. Marcus A, Chen E (2002) Designing the PDA of the future. interactions, 9 (1): 34-44
37. Maulsby D, Greenberg S, Mander R (1993) Prototyping an intelligent agent through Wizard of Oz. In: SIGCHI conference on Human factors in computing systems, Amsterdam, The Netherlands, pp 277-284.

38. McDougall SJ, de Bruijn O, Curry MB (2000) Exploring the effects of icon characteristics on user performance: the role of icon concreteness, complexity, and distinctiveness. Journal of Experimental Psychology, 6 (4): 291-306
39. Miller GA (1956) The magical number seven, plus or minus two: Some limits of our capacity for processing information. Psychological Review, 63: 81-97
40. Mills CB, Weldon LJ (1987) Reading text from computer screens. ACM Computing Surveys, 19 (4): 329-358
41. Moffett SE, Menon AS, Meites EM, Kush S, Lin EY, Grappone T, Lowe HL (2003) Preparing doctors for bedside computing. Lancet, 362 (9377): 86
42. Myers B, Ioannidis Y, Hollan J, Cruz I, Bryson S, Bulterman D, Catarci T, Citrin W, Glinert E, Grudin J (1996) Strategic directions in human-computer interaction. ACM Computing Surveys (CSUR), 28 (4): 794-809
43. Myers BA (2001) Using handhelds and PCs together. Communications of the ACM, 44 (11): 34-41
44. Nielsen J (1990) Paper versus computer implementations as mockup scenarios for heuristic evaluation. In: IFIP TC13 Third Interantional Conference on Human-Computer Interaction, pp 315-320.
45. Nielsen J (1993) Usability Engineering. Morgan Kaufmann, San Francisco
46. Nielsen J (1994) Estimating the number of subjects needed for a thinking aloud test. International Journal of Human-Computer Studies, 41 (3): 385-397
47. Norman DA, Draper S (1986) User Centered System Design. Erlbaum, Hillsdale (NY)
48. Oquist G, Goldstein M (2003) Towards an improved readability on mobile devices: evaluating adaptive rapid serial visual presentation. Interacting with Computers, 15 (4): 539-558
49. Pace BJ (1984) Color Combinations and Contrast Reversals on Visual Display Units. In: Conference Proceedings of the Human Factors Society, Santa Monica (CA), pp.
50. Peirce CS (1932) Collected Writings on Semiotics. In: Hartshorne C, Weiss P (eds) The Collected Papers of Charles Sanders Peirce 2, Harvard University Press, Cambridge (MA), pp. 135ff
51. Rettig M (1994) Prototyping for tiny fingers. Communications of the ACM, 37 (4): 21-27
52. Reuss E, Menozzi M, Buchi M, Koller J, Krueger H (2004) Information access at the point of care: what can we learn for designing a mobile CPR system? International Journal of Medical Informatics, 73 (4): 363-369
53. Shackel B (1986) Ergonomics in design for usability. In: HCI-86: People and Computers II: Designing for Usability, York (UK), pp.
54. Shneiderman B (2002) Leonardo's Laptop: Human Needs and the New Computing Technologies. MIT Press, Boston (MA)
55. Shneiderman B, Plaisant C (2004) Designing the User Interface 4th Edition. Pearson, Harlow (UK)
56. Shubin H, Falck D, Johansen AG (1996) Exploring color in interface design. interactions, 3 (4): 36-48
57. SIGGRAPH (1999) Physiological Principles for the Effective Use of Color. Online at: http://www.siggraph.org/education/materials/HyperGraph/color/coloreff.htm (last access: 20030908)
58. Simon HA (1974) How big is a chunk? Science, 183: 482-488

59. Stary C, Peschl MF (1998) Representation Still Matters - Cognitive Engineering and Task-Based User Interface Development. Behavior and Information Technology, 17 (6): 338-360
60. Stary C, Vidakis N, Mohacsi S, Nagelholz M (1997) Workflow-Oriented Prototyping for the Development of Interactive Software. In: IEEE COMPSAC 1997, pp 530-535.
61. Stephanidis C, Salvendy G, Akoumianakis D, Arnold A, Bevan N, Dardailler D, Emiliani PL, Iakovidis I, Jenkins P, Karshmer A, Korn P, Marcus A, Murphy H, Oppermann C, Stary C, Tamura H, Tscheligi M, Ueda H, Weber G, Ziegler J (1999) Toward an Information Society for All: HCI challenges and R&D recommendations. International Journal of Human-Computer Interaction, 11 (1): 1-28
62. Sutcliffe A (2002) User-Centred Requirements Engineering: Theory & Practice. Spinger, Berlin, Heidelberg, New York
63. Tauscher L, Greenberg S (1997) How people revisit web pages: empirical findings and implications for the design of history systems. International Journal of Human–Computer Studies, 47: 97-137
64. Tullis TS (1997) Screen Design. In: Helander M, Landauer TK, Prabhu P (eds) Handbook of Human-Computer Interaction, Elsevier, Oxford, pp.
65. usabilitynet.org (2003) ISO 13407. Online at: http://www.usabilitynet.org/tools/13407stds.htm (last access: 20030611)
66. Virzi RA (1992) Refining the test phase of usability evaluation: how many subjects is enough? Human Factors, 34 (4): 457-468
67. VisiBone (2003) Webmasters Color Laboratory. Online at: www.visibone.com (last access: 20030922)
68. Young PMC, Leung RMW, Ho LM, McGhee SM (2001) An evaluation of the use of hand-held computers for bedside nursing care. International Journal of Medical Informatics, 62 (2-3): 189-193
69. Ziefle M (2001) User productivity and different screen technologies: CRT screens with high refresh rates versus LCD displays. The Display Search Monitor, 6 (14): 11-14