

Usability Study of a WAP Service

The Art of Overcoming Constraints

© 1999-2000 IPLab & Ericsson
Business Networks AB

Helge Hüttenrauch, hehu@nada.kth.se
David Bezjak, b96_bed@b.kth.se
Jesper Larsson, v96_lar@v.kth.se
Daniel Lennmalm, b95_lal@b.kth.se
Anders Lundin, v96_lus@v.kth.se

1	ABSTRACT	4
2	INTRODUCTION	4
3	WAP – WIRELESS APPLICATION PROTOCOL	5
3.1	INTRODUCTION TO WAP	5
3.1.1	<i>Technical Overview</i>	5
3.2	WAP – HERE TO STAY OR JUST A HYPE?	6
4	PERSONAL SCREEN CALL SERVER – PSCS	7
4.1	INTRODUCTION TO PERSONAL SCREEN CALL SERVER	7
4.2	AVAILABLE SERVICES	8
4.2.1	<i>Scheduler</i>	8
4.2.2	<i>Directory</i>	8
4.2.3	<i>Message Diversion</i>	9
4.2.4	<i>Call Log</i>	9
4.2.5	<i>Personal Number</i>	10
4.2.6	<i>Call Handler</i>	10
5	USERS	12
5.1	DEFINITION	12
5.2	USER SITUATIONS	12
6	PROBLEM IDENTIFICATION	13
6.1	MAKING THE USER FEEL FAMILIAR	13
6.2	GRAPHICAL INTERFACES	13
6.3	STANDARD.....	14
6.4	MENUSYSTEM.....	15
6.4.1	<i>Design</i>	15
6.4.2	<i>Structure</i>	15
6.5	HELP SYSTEM	16
7	INITIAL USER STUDY	17
7.1	TESTPANEL	17
7.2	PURPOSE AND IMPLEMENTATION	17
7.3	FACTORS THAT MAY HAVE INFLUENCED THE TESTPANEL.....	17
7.4	EVALUATION	17
7.4.1	<i>How is the PSCS used today</i>	17
7.4.2	<i>What do the users wish for in the PSCS</i>	18
8	DESIGN OF SERVICES	19
8.1	INTRODUCTION	19
8.2	STARTING POINT	19
8.2	SEARCH (DIRECTORY)	19
8.2.1	<i>Changes in Terminology</i>	19
8.2.2	<i>Alternative 1</i>	19
8.2.3	<i>Alternative 2</i>	21
8.2.4	<i>Alternative 3</i>	21
	<i>Output</i>	22
8.4	PERSONAL NUMBER	23
8.4.1	<i>Select Profile</i>	23
8.4.2	<i>Define Number</i>	24
8.4.3	<i>Add Profile</i>	25
8.4.4	<i>Edit Profile</i>	25
8.4.5	<i>Delete Profile</i>	26
8.4.6	<i>Settings</i>	26

9	SECOND USERSTUDY ITERATION	28
9.1	TESTPANEL	28
9.2	PURPOSE	28
9.3	IMPLEMENTATION	28
9.4	FACTORS THAT MAY HAVE INFLUENCED THE TESTPANEL	30
9.5	EVALUATION AND REDESIGN	31
	9.5.1 <i>Search (Directory)</i>	31
	9.5.2 <i>Personal Number</i>	35
10	THIRD USERSTUDY ITERATION	42
10.1	ABOUT THE USERSTUDY	42
10.2	EVALUATION AND REDESIGN	42
	10.2.1 <i>Search (Directory)</i>	42
	10.2.2 <i>Personal Number</i>	44
11	CONCLUSIONS	47
11.1	REFLECTIONS	47
11.2	ABOUT THE APPLICATION	47
11.3	HOW TO PROCEED	48
12	APPENDIX	49
	APPENDIX A	49
	<i>Abbreviations</i>	49
	<i>Definitions</i>	49
	APPENDIX B	50
	APPENDIX C	51
	<i>WLM application code tried</i>	51
	APPENDIX D	52
	<i>Tasks to perform during the userstudy</i>	52
	APPENDIX E	53
	<i>Introductory userstudy 99-11-17</i>	53
	APPENDIX F	54
	<i>Final Result in Overview</i>	54

1 Abstract

This report describes a Wireless Application Protocol (WAP) usability study conducted upon selected Personal Screen Call Server (PSCS) based functions. A short introduction to WAP, PSCS, the intended user group, and our methods are given. We then identify the problems such as different terminals and standards encountered.

Three user studies¹ or design iterations upon the menu-design of the Personal Number (PN) and a telephone-directory search are described in detail. The user observations and comments are discussed. Technical limitations are not taken up to focus upon usability and not implementation. As an example of WAP enabled terminals the Graphical User Interface (GUI) of the Ericsson mobile phone RS320 was selected due to available guidelines. Finally, we present our conclusions for further investigation and development.

2 Introduction

This project started off as part of a course in Human Computer Interaction (HCI) at The Royal Institute of Technology (KTH) in Stockholm, Sweden. Co-operation of this kind has been asked for both from industry and the academic world. As it is not yet common, this project might be regarded as a pilot project to be evaluated afterwards. These types of co-operations are more often recognized during the degree projects, but this course with its "problem based learning" approach gave the students a chance to experience the "reality" at an earlier stage during the education.

Involved partners in this project are the Interaction and Presentation Laboratory (IPLab), represented through Helge Hüttenrauch as course supervisor, and Ericsson Business Networks AB, namely Bruno von Niman, Usability Expert, Erika Näslund, Engineering, and Henok Welday, Engineering.

As none of this project experience would have come true without the engagement and encouraging support of Bruno, Erika, and Henok, we are greatly indebted to them and would like to express our gratitude of giving us the chance to perform this study. We hope we can reward them with the findings of this report and future co-operations!

The goal is to find out how a user-friendly WAP-application based upon Ericsson PSCS² is possible to develop. This report describes the different project parts of the design process in chronological order. The task is structured along the general parts of product/service selection, technology and problem introduction and identification, design sketch and iterated usability design and evaluation cycles.

Readers familiar with WAP or PSCS could skip the 3rd resp. 4th chapter and directly start out with the user group description in chapter 5. Main problems are described in chapter 6. They are tried to solve in the different iterative design cycles in chapter 7 through 10. Conclusions are summarized in short form in chapter 11. For a more detailed information of the different materials used, please refer to the different appendix-parts.

¹ See Appendix D and E

² See screen dumps from Ericsson's PSCS in chapter 4, (taken: 991201)

3 WAP – Wireless Application Protocol

3.1 Introduction to WAP

Twenty years ago many people had never used a computer, ten years later the same phenomena was true for the usage of mobile phones or the Internet. Nonetheless are we today facing the convergence of the mobile phone connected to the Internet. Today many people are dependent on communication devices and services. They use them every day, both for private usage and in to accomplish their work tasks. The technologies involved have had an explosive development during the last years and an end to this development is not in sight. People today need to be mobile, the businessmen/man of today works flexible at different places, both on-site as well as off-site. This sets high demands on communication infrastructures and technologies. The need to communicate while being mobile drives the development of new services. We are arguing, in short, for a convergence of the world of mobile communication and the Internet in developments to come.

WAP³ stands for Wireless Application Protocol, and is a new technology developed in co-operation between Ericsson, Nokia, Motorola and Phone.com, (former Unwired Planet). WAP is an open global specification that enables users to access web-based interactive information-services and applications from the screens of their mobile phones or WAP enabled terminals. These include both consumer and corporate solutions like e-mail, news, information services, entertainment, electronic commerce transactions, banking services etc. The purpose of the WAP de facto standard is to specify the ways in which the IP based Internet and Intranets are linked to GSM and other networks.

WAP defines both a protocol and a language. The protocol contains rules governing the ways in which information is to be transmitted. The language used is WML⁴, and fulfils the same function as HTML does for World Wide Web browsers.

3.1.1 Technical Overview

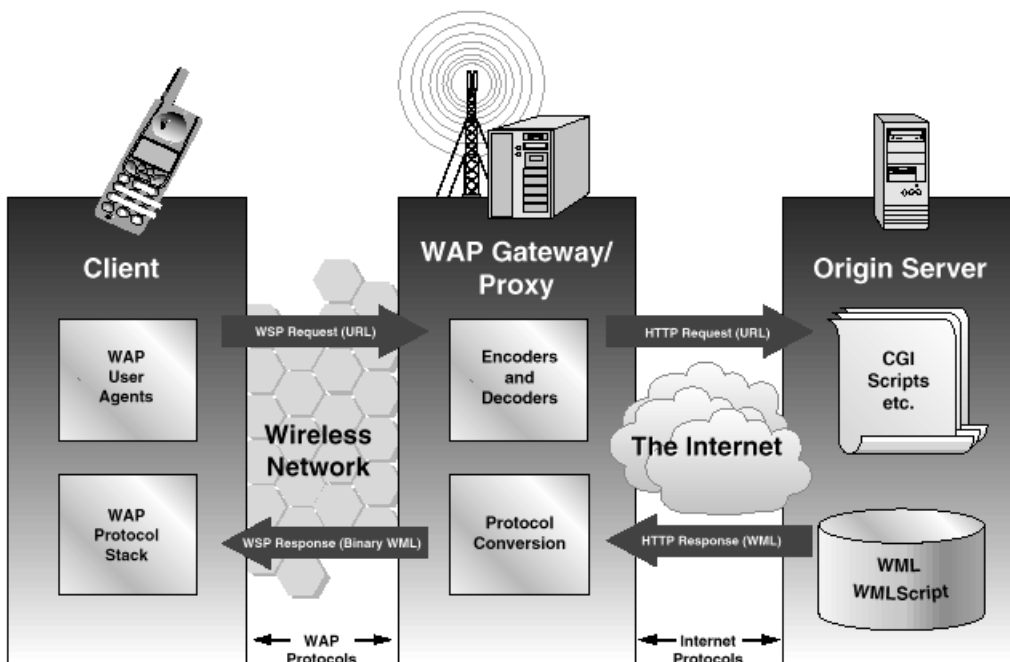


Figure 3.1. The WAP Model⁵

³ For explanation of terms see: Appendix A

⁴ For more information about WAP and WML see: <http://www.wapforum.org/>

⁵ WAP White Paper, February 1999

WAP is a standard for communication between a WAP server (or a WAP gateway), that is connected to the Internet, and a small hand held device with a limited GUI.

It includes:

- A markup language, WML, which is similar to HTML, but with a smaller grammar set and several other advantages (and shortcomings) listed in the wireless markup language specification.
- A scripting language called WMLScript similar to JavaScript.
- A bundle of telephony related functions called WTAI that makes it possible to add a name to the client's phonebook or send SMS text.

3.2 WAP – Here to stay or just a hype?

“Technology adoption has been slowed by the industry's lack of standards to make hand-held products Internet compatible”⁶. This indicates that a standard like WAP has been needed for a long time. Now a critical mass of manufactures (see above) has joined this standard and it will probably grow fast which is one of WAP's problems. A fast growth often means problems with standards, because the standard committee doesn't keep up with the development.

Another problem is that Wireless Transport Layer Security Specification (WTLS) does not give enough security. “There is a gap and it lies within the Operators Network. The translation of binary compressed information to HTTP happens there and theoretically it could be intercepted by Network employees or by compromising the networks security”. Another important aspect is given as, “Maybe we just have to wait until encryption can be done using the SIM-card. This way the content would be secure all the way”.

This report will not discuss WAP and mobile Internet security, but it definitely raises some questions about the WAP technology future. Security aspects will be an important issue in the coming development of WAP, because this is a critical part of the protocol, many different interests have to be fulfilled, and compromises might be unacceptable.

If today's usage context would not change during the next years, WAP would probably die. The price for using WAP services is the same as for GSM phone-calls. At the same time the bandwidth that can be used to send data over the WAP-protocol is limited to 9.6 kbps. But with the General Packet Radio Service (GPRS) and compatible terminals coming, availability is foreseen in 2000/2001 in selected networks, the overall situation will change. “GPRS is a packet-based wireless communication service that, when available in 2000, promises data rates from 56 up to 112 kbps and continuous connection to the Internet for mobile phones”. This could mean less and shortened dialup-time as well as staying online for a longer time. Additionally the opportunity to switch to a different charging system is made possible. One will only be charged for the amount of data received and not for the time connected. At the same time the user will have a continuous connection which leads to new possibilities in service offerings.

To sum up the discussion and answer the question whether WAP is to be regarded as a hype or solid technology to be taken seriously, we have looked at the coming technologies discussed. It is our opinion that the WAP standard will, if development goes as planned, continue to grow and prosper even though there are problems that have to be addressed, discussed, and solved. The hype on the (financial-) markets and in media have to be seen in this light. The urge to produce these services is enthusiastic, and the impatience created has often been expressed in seemingly explaining phrases like “WAP stands for ‘Where Are the Phones?’”. We believe that they and the services will come.

⁶ <http://www.wapforum.org> (accessed: 991203)

4 Personal Screen Call Server – PSCS

4.1 Introduction to Personal Screen Call Server

PSCS provides server-client services developed to make the daily use of advanced telephone services easier. The PSCS is a TCP/IP based client application allowing different Web-browsers to access the services within the Intranet of a company for example. PSCS was developed by Ericsson Business Networks AB for usage together with their MD110, an advanced digital telephone switch (often also referred to as a PBX). Using PSCS services, users are enabled to initiate and administer their available telephony services through a screen based browser application, additionally to their digital stationary or mobile phone.

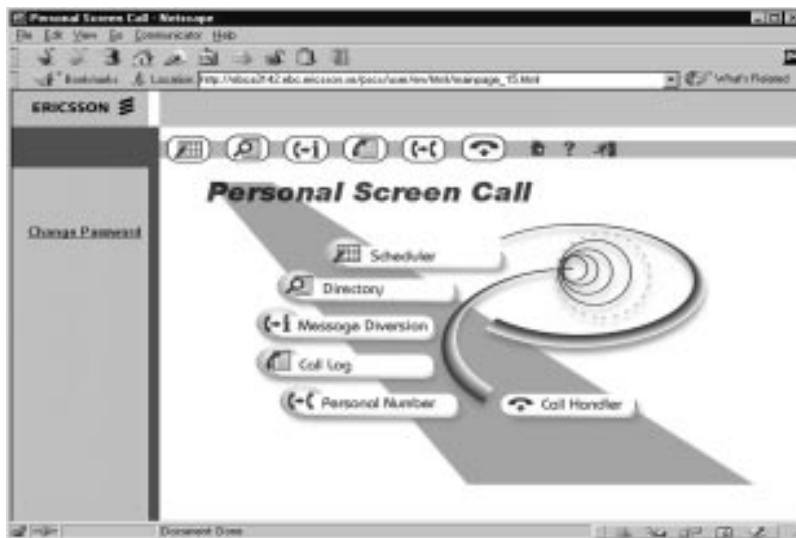


Figure 4.1. PSCS start page

As an introduction to PSCS some main functions are described below. They are meant to serve as a background against which we will design our WAP functionality beginning in chapter 8 (for more detailed information, please be referred to the PSCS and related services' documentation). Keywords used are often taken directly from that application, therefore being familiar with PSCS will help in understand our design(s). We would like to explicitly point out a fact easily not noticed because it is so intuitive:

PSCS is a application run on a standard computer screen. Representation of services and interaction follow the Internet Browser and the interface design and interaction guidelines of so called WIMP (Windows, Icons, Mouse, and Pointers) interfaces. This is important to highlight here, since PSCS can be seen as a representative of a class of existing application and services with a certain characteristic "flavor". It is this character which will need to change going over to WAP for different reasons, as we will see shortly.

We are not analyzing the services and interaction of a mobile phone in a similar way to stay focused. However, we assume basic knowledge of terminology and handling at least for Ericsson mobile phones. For further reference, please also consult for example the White Pages/WAP-Guidelines on the Ericsson RS320 model.

4.2 Available services

4.2.1 Scheduler

“Scheduler” is working somewhat like a “filofax for the phone”, where the user is able to pre-set a schedule to perform different telephony service tasks. The user is for example given the possibility to be reminded to call a number at a certain time, to divert the phone to another number during a specified time-period and/or activate either “Message Diversion” or “Personal Number”.

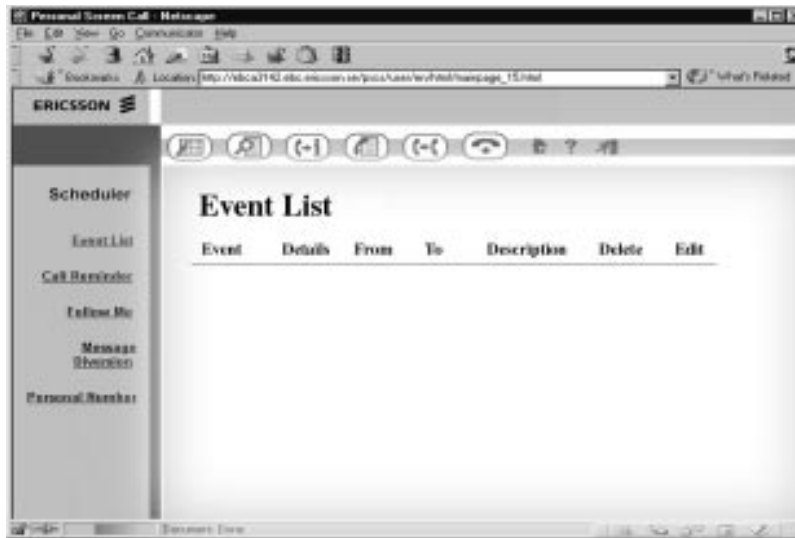


Figure 4.2. PSC Scheduler

4.2.2 Directory

The “Directory”-service is used to search for information in a company (telephony) directory such as phone number and e-mail concerning employees within a company’s database. The service also provides a possibility for the user to set up and edit an own phonebook. This specific service is called “Private”. The actual search operation can be performed in both directories, the company database as well as the own phone-book. The search result is presented as first and last name, phone-number, department, and e-mail address.



Figure 4.3. PSCS Directory

4.2.3 Message Diversion

The “Message Diversion”-service enables the user to inform a caller that he or she is busy at the moment with some kind of activity. The information is presented to the caller in two different ways. If it is an internal call a message will appear in the telephone display, e.g. “meeting 15:00”. If it is an external caller they will be connected to an operator, who can convey the message and react accordingly. The advantage is that a user can tell the caller both the cause of not being available and when another try to communicate is expected to be more successful.



Figure 4.4. PSCS Message Diversion

4.2.4 Call Log

The “Call Log”-service automatically stores two different lists containing call related information. One list shows incoming, the other one outgoing calls. The numbers are presented together with information about the time and date of the call. If an incoming call is internal, also first and last name of the caller is logged. If the user has been absent from the office, he or she can easily get information about missed calls after returning to the office by checking the call log.



Figure 4.5. PSCS Call Log

4.2.5 Personal Number

“Personal Number” provides different services, giving the user the possibility to customize a so called Personal Number. The main idea is to substitute the many telephone numbers one often uses in business/workplace communication (e.g. voice-, fax-, mobile-, home- etc. telephone) with only *one telephonenumber* under which the subscriber can be contacted. The Personal Number service makes it possible to administrate the intelligence behind this system, e.g. administration to which fax-number an incoming fax on the personal number should be forwarded to. With the aid of this service the user can therefore set up and edit what is called a “profile”. The profile specifies at which number the user can be reached, in which order the different numbers should be paged, (e.g. mobile, office or voicemail), and for how long the number should be paged.



Figure 4.6. PSCS Personal Number

4.2.6 Call Handler

The “Call Handler”-service provides the user with the possibility to use the application interface to administer his/her calls instead of the physical phone interface. The PC can be used to dial a desired number

or to take calls by activating the speakerphone from the application. If the user has selected the “Directory”-service to get hold of a phone-number, (part 4.2.2), he or she can simply double-click on the number returned, and that way start the “Call Handler” to dial the number

5 Users

5.1 Definition

Designing a useable interface, we have raised the question of who will actually benefit from having a WAP based telephony service such as the Personal Number or the Directory Search. According to Bruno von Niman, usability expert at Ericsson Business Network, the targeted user group could be described as working in "mobile office", a term used to express their high need for physical mobility to perform their work. This group with high mobility and communication needs is also referred to as the "Generation X". The Generation X is represented by users 20-35 years of age, which are accustomed to a completely different way of working when compared to the traditional office working conditions. They are heavily relying and depending on the possibility to be able to get information from their company's Intranet or Internet. This makes it possible for them to fulfill their communication needs, e.g. when working from at home (off-site), being at customer sites or out travelling.

Costs for the service can thus be assumed to be secondary for the foreseen user group as it will cost more for the user/organization he/she is working for, to not be able to count on the benefits of using the services offered.

We can also deduct that the Generation X user of the WAP-services foreseen is highly educated and trained in usage of existing communication technologies and not afraid to take up new devices and technologies, if the benefit becomes obvious to them. These users will be pioneering in taking up the WAP technology in society.

5.2 User Situations

One of the thoughts with the WAP-application of the PSCS is to provide the Generation X users with a possibility to handle unforeseen situations in an easy and flexible way. We believe that the user praise efficiency for increased comfort and flexibility and therefore will use the best suited information and communication service and technology being available at any given moment. This in turn means, that they will *not use the WAP application when a better alternative is available*. The WAP application is therefore very dependent upon finding its *niche of acceptance* providing an added value in the moment of using the service.

It is furthermore important, due to the current WAP technology or terminal limitations, to not only provide the right service(s) at the right time, but also to provide *exactly the needed functionality in very compact form*. We will therefore assume the user to be familiar with the both the service itself (and even the WAP implementation) before it will actually be used. This requires the user to master the learning curve associated with using the WAP service(s) before they are actually needed for real. The user will then *know what to expect and be familiar with the reduction of service offerings* due to the WAP limitations.

With the specified user-group in mind we could set up a few scenarios serving as examples: Imagine a user who got a travel-order on short notice. The flight/train leaves in 45 minutes, putting stress upon the user to get his/her stuff together. To divert the phone might not be the first thing that comes to mind. Maybe there is simply no opportunity to divert the phone. With a WAP-implementation of the Personal Number functionality, the user could get an opportunity to perform this task for example on the way to the airport.

Another scenario could be a user entering a meeting directly in the morning, without the opportunity to first stop by in the office just to divert the phone. With the WAP-application the user would be able to take care of this matter in advance, for example on the way to the meeting. This could save time and make things more convenient for the user.

6 Problem identification

6.1 Making the User feel familiar

The hardware and software ability to show the user relevant information with a GUI differs greatly between a PC-application and a WAP-application. The large difference in size of the screen used in both cases is obvious. A “large” computer screen competes against a rather small one on the different WAP enabled portable devices, e.g. a GSM-phone (fig. 6.1). With this condition in mind, the question is how to create an interface, so that a user who is familiar with the PC-browser based PSCS functionality will be able to make the transition to a small screen WAP-application. The WAP-application should be built using the same structure as the one used for the PC-application as long as possible. Another important aspect is to try to use the same terminology.

6.2 Graphical Interfaces

A quick look at the WAP supporting hardware on the market shows a wide number of different designs when it comes to the interface. Most of the time it is the size of the interface that differs a lot, but it is equally important to attend the differences in the input modalities offered (keyboard, pen on a touch sensitive screen, keypad), the overall size, format of the screen, how the device is physically placed during main operation (one-hand, both hands, lying on a table etc.), other available (for example build-in) applications to name but the most important attributes. Looking at the three different WAP supportive hardware from Ericsson, soon to be released or already on the market illustrates the different classes of hardware to expect.



Figure 6.1. Ericsson 's WAP supporting hardware. Three different platforms, sizes, and interfaces

Due to the different abilities of the hardware WAP system developers are either destined to create a flexible system that is possible to apply under different conditions or create target-platform specific services. The main question ”How to create an interface that is able to show the information necessary?” needs to be answered. We believe that one will have to set a priority of which platforms to support, what to show, when and how to show it. These design choices ought to be based upon the available design possibilities

and suited interaction modalities. It should be explored, specified, and tested in a User Centered Design (UCD) method, cooperatively collecting the users' needs to maximize usability.

6.3 Standard

There are different standards to look upon designing WAP applications. The different types of hardware to be supported are likely to have their own guidelines or standards. They are often coupled both to the operating system (OS), e.g. Palm OS or EPOC, as well as the vendor's specific GUI and interaction. One crucial question is if there is a common standard for WAP applications between different manufacturers of WAP enabled hardware and software. In other words, will the applications really work on different manufacturer's hardware? We assume that some kind of interworking comparative testing is performed by the WAP-forum and associated companies, but have in fact no clear knowledge that this actually the case. One definite problem encountered (at least in the available Software Development Kits) is that equally marked keys on different platforms actually perform different tasks in one application. Equally, the graphical output on the available screen might look different on different vendors' platforms. This will have consequences e.g. when users are prompted for input, but the sequence looks differently and requires different input actions dependent upon the vendors' name on the terminal. Can it be that one has to develop different versions of the application in order to be able to use it on different manufacturer's hardware? This would be a great setback for WAP as a technology.



Palm V



Nokia 7110



Siemens s25

Figure 6.2. Different hardware

The above mentioned adds to the complexity of designing one WML-code that will work on all user agents. The physical prerequisites are shown as well – screen resolutions differ for example between the Ericsson R320 (101x65 pixels) and the Ericsson MC218 with a resolution of 640x200 pixels. It will be very challenging to design a WAP-service with only one WML-code that suits all user agents. Challenging does nevertheless not mean impossible as there is a way of customizing what is sent to the user agent depending on its characteristics. The user agent here sends its characteristics to the WAP origin server/gateway and an appropriate bytecode for that particular GUI is sent to the user agent⁷. This will generate additional work for the application developers and make development of WAP-services less interesting. To sum up the discussion it is obvious that there is even more standardization needed with the

⁷ see “WAP WAE Specification version 24-May-1999 5.1.4”

WAP-technology as it has barely reached a maturity plateau. Future versions of the WAP will provide for possibilities to address many of the issues raised.

6.4 Menu system

6.4.1 Design

In order to design a usable interface, one should try to build a menu system with a uniform structure, and strive to make similar parts of the application look and feel alike for consistency⁸. This will help the user to recognize what he/she sees in different parts of the application. For example the same type of input-field ought to be used every time some kind of input to the system needs to be performed. “Radio-buttons” (fig. 6.3) ought to be applied when the user has a single choice and “checkboxes” when a multiple choice needs to be presented. These guidelines follow established usability guidelines and standards⁹.



Figure 6.3. “Radiobuttons” and “Checkboxes”

6.4.2 Structure

In order to achieve a consistent and consistent structure in the application, one has to think about which kind of structure to use when defining the application, including the grouping and hierarchical substructuring between different parts/steps of the application. This is mostly to prevent the application from behaving different in similar situations, which will confuse the user.

One way to approach this is to use well-defined, hierarchical relations with strict choices between different menus, (fig. 6.4a). A structure designed like this makes it easy for the user to keep track of the location in the menu (for navigation), but the time it takes to perform certain tasks might be extended due to the stringent sequencing required. For example if the user wants to move “sideways” between different main menu functionality blocks. The user then first has to “move up” to a higher navigation level before being offered to switch to a different main menu function and being able to click him/herself “down” again. This kind of navigation is very efficient in small information/selection spaces to be navigated, or if the navigation itself is not very easy to perform, e.g. due to limited visual feedback.

Another approach is to have the menus hyperlinked together so that they would be accessible in different ways, (fig. 6.4b). For the user, this could in some cases result in shorter interaction times when moving between different parts of the system, but at the same time it makes navigation more difficult since the probability to lose track of the current position is increased at least for novice users.

⁸ Shneiderman, Ben, *Designing the Use Interface* (Third Edition), 1998, Chapter 7.2

⁹ Design Guidelines for R320, <http://www.ericsson.se/WAP/developer/Tools/index.asp> (991203)

7 Initial User Study

7.1 Testpanel

This study was limited to a total of 18 users¹⁰ employed by Ericsson. They were more or less familiar with and using the PSCS services. The choice to only study "internal" users was dictated mainly by the time constrains. It would have been preferable to work with employees of a company outside of Ericsson that uses the PSCS solution. The testpanel consisted of equally male and female gender. The age range was not actively considered, but we feel that the persons performing our tests fell into the age range given above for the Generation X users. However, they did not (yet?) fulfill the requirements in need for mobility and communication, as far as we could tell. We foresee the difficulty to really gather representatives fitting well into the Generation X profile. According to definition, they are very busy people, so to schedule a meeting for usability testing purposes might require serious pre-planning on a large time and financial scale.

7.2 Purpose and implementation

A brief user study was performed to get a picture of the PSCS services most frequently used. This was to create a basis for the further development of the WAP-application and to validate the importance of the Personal Number and Directory Search service.

The study has been conducted as interviews, both by phone and in person, where the testpanel have been asked a number of different questions¹¹. This method allowed for comments, ideas, and thoughts to be raised, i.e. a qualitative data collection. As a shortcoming of the existing GUI, e.g. missing options were mentioned.

7.3 Factors that may have influenced the testpanel

Because PSCS is a service developed by Ericsson and the testpanel are Ericsson employees, we need to assume biased opinions having effect on the answers given. We mark this down as a clear shortcoming in our study methods and would therefore propose the validation of our findings with an Ericsson external, in size statistical relevant, and representative user group.

7.4 Evaluation

7.4.1 How is the PSCS used today

The study has shown that only some of the services available in the PSCS are actually used. The "Personal Number"-service, (part 4.2.5), was used by all the users studied. The "Directory"-service, (part 4.2.2), is used by a few, while the other available services are sparsely, or not used at all.

The "Personal Number"-service, which is one of the most frequently used services, is in almost all cases operated from DECT mobile phones (Freeset) and only sometimes from the PSCS-application. In the case where it is used from PC workstation the purpose is most of the time to set up or edit different profiles. While from phone it is mostly used to select and set one of the existing profiles. When talking about using "Personal Number" from a GSM phone, most of the users think it is too complicated to handle, since the user is supposed to enter over thirty (sic!) digits in a correct way in order to select a profile. Non of the studied users has actually done so, it's simply unusable.

¹⁰ 4 users in the initial study, 7 users in the 2nd and 3rd usability iteration.

¹¹ see Appendix E

7.4.2 What do the users wish for in the PSCS

The most outspoken wish from the users is to be able to divert incoming calls to different numbers in an easy and flexible way. Some of the users also raised some more specific wishes, like being able to set and change the time specifying for how long the phone will ring at a certain number before further diversion. A major concern for simplicity was raised repeatedly by different users. The task is to create an interface that is simple to work with, but still provides the user with the necessary options and services that he/she needs.

We found that users' attitude are positive to using a WAP-application of the PSCS functionality (if it would be available to them), as long as it is simple and easy to use.

8 Design of services

8.1 Introduction

We decided to look at the services “Directory” and “Personal Number” and started by making some sketches of different proposals for the design of the menu system. The different proposals were designed according to functions supported by the Ericsson R320¹². Furthermore we tried to take in the users’ considerations for simplicity. The terminology was taken from the PSCS interface.

As a general balancing we found the different solutions having advantages and disadvantages. It is e.g., often difficult to decide for or against a certain GUI representation and the associated keypad-pressing sequences necessary. Some nice GUI representation might require more key-presses to complete the task than a different graphical approach, judged to be inferior to the other alternative graphically, but providing a better keypad-interaction sequence to complete the task.

8.2 Starting point

With the knowledge of the PSCS terminology and the different idea sketches we started to develop different realistic proposals for the WAP- application in mind. With these proposals another user study was then performed to get feedback on the design and to improve the solution. This procedure was iterated and supported us to design the solution for our final proposal.

8.2 Search (Directory)

8.2.1 Changes in Terminology

We have decided to change the name of this service to “Search” instead of “Directory” because it is more informative according to our user study. This change gives some new terminology for a user that is familiar with PSCS and we hope that “Search” is informative enough. We have decided not to use the “Private”-service for the WAP-application because nobody in our user study uses this service, and we do not think other users will either.

8.2.2 Alternative 1

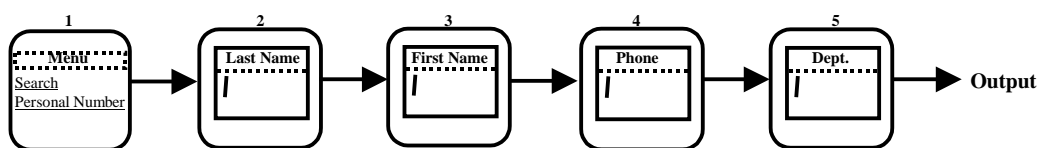


Figure 8.1. Search menu, (alternative 1).

General idea(s)

The idea for this proposal was to get a consistent and easy input method. When the user chooses to continue from an empty card (see WML-terminology for reference) the search-topic will not be used in the actual search. Whether the user has entered an input or not, the “YES”-key is used to move on to the next card. This will minimize the risk of mistakes since it is easier for the user to remember which key to press¹³.

We think this solution should be fast and appreciated by a user familiar with the search-topics available. This solution also offers the user the possibility to go back and make changes in the previous input-field

¹² Design Guidelines for R320

¹³ Shneiderman, Ben, *Designing the Use Interface*, Chapter 2.8

by using the “NO”-key. The most favorable for this alternative is that it uses least key-presses of the three alternatives.

A disadvantage with this alternative could be that an unfamiliar user neither knows how many input-fields will be shown, nor which topics that are available to search on. This might confuse a user.

Keystroke-level model

We have tried to model which keys need to be pressed by introducing what we call a ”Keystroke”-modeling. The idea behind this is to be able to compare the necessary keystrokes with the output on the screen. The number and kind of keys pressed are given, thus providing the chance to evaluate not only the absolute number of keystrokes needed, but the variety of multiple keys to be pressed in a certain sequence. Only if looked upon in this combination will a usability judgement make sense. Other factors like familiarity with the Ericsson ”Yes-No” navigation system for example are not considered here. The user input of strings, e.g. in names, are only counted as one keystrokes.

Helge pointed out other standardized methods of getting usability heuristics on keystrokes, but advised on not using them due to increased complexity not fitting the project’s time frame.

The model only counts keystrokes for error-free performance, but is based on the model by Card, Moran and Newell¹⁴.

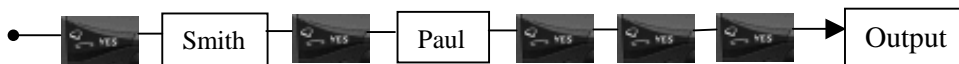


Figure 8.2. Search for “Paul Smith” gives 7 key-presses¹⁵.

¹⁴ Shneiderman, Chapter 2.2.2

¹⁵ “Paul Smith” counts as two key-presses

8.2.3 Alternative 2

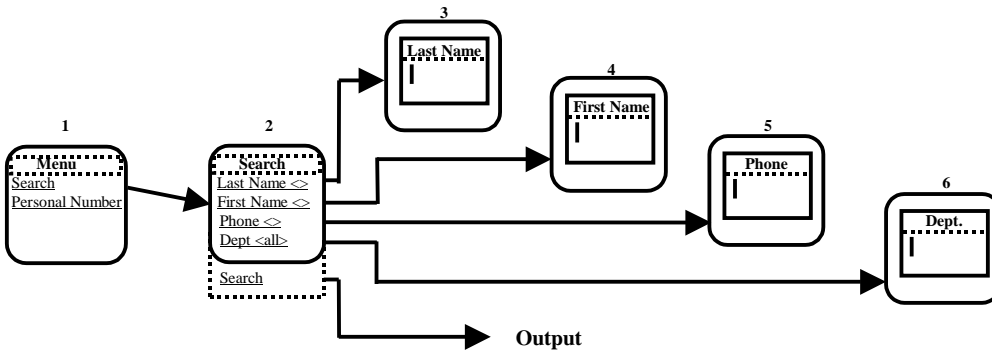


Figure 8.3. Search menu, (alternative 2)

General idea(s)

This design was developed to give the user an overall picture of the different search-topics, and at the same time show previous inputs (part functions). In the “Search”-card, (fig. 8.3:2), the user has the option to choose different search-criteria. Selecting a highlighted option results in an input-field, (fig. 8.3:3-6). When choosing the “Search”-option, entered criteria are used for the actual search. This design gives the user a clear picture of the search-topics available.

One negative aspect could be that the design demands more key-presses than alternative 1, which might be irritating for a user that knows what this service provides.

Keystroke-level model

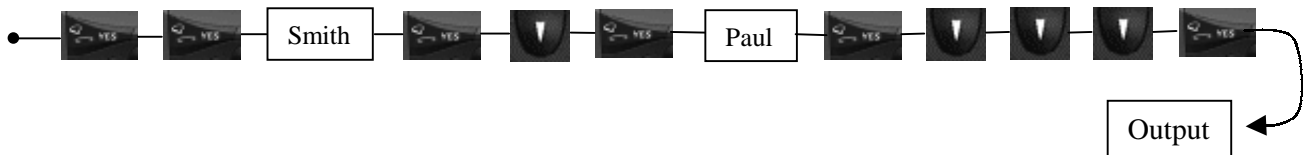


Figure 8.4. Search for “Paul Smith” gives 12 key-presses.

8.2.4 Alternative 3

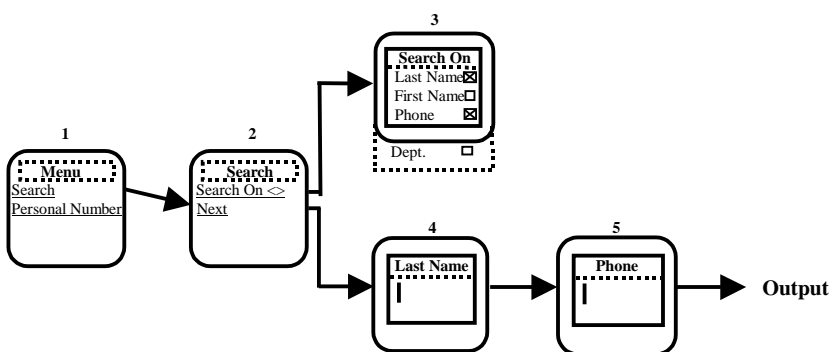


Figure 8.5. Search menu, (alternative 3)

General idea(s)

This design allows the user to specify which search-criteria to enter, (fig. 8.5:3). Input-fields related to the topics ticked in the checkbox are then showed, (fig. 8.5:4-5).

For a new user this design gives a good picture of the topics available. The frequent user might think it is too lengthy to work with. At the same time the number of key-presses needed enhances the risk of errors.

Keystroke-level model

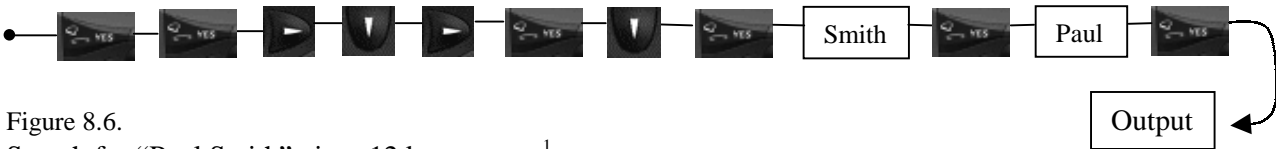


Figure 8.6. Search for “Paul Smith” gives 12 key-presses¹.

Output

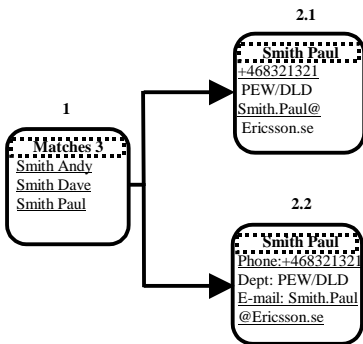


Figure 8.7. Output menu, (two alternatives)

General thoughts

The number of matches after the search is shown in the display headline. The matches are presented in alphabetic order by last name. Selecting a highlighted option will result in a new card showing additional information about the person. The information may be displayed as showed in fig. 8.7:2.1 or fig. 8.7:2.2.

The alternative showed in fig. 8.7:2.1 demands least space, with fewer row-breaks as result. The disadvantage could be that the search result might be misinterpreted.

The alternative shown in fig. 8.7:2.2 was designed to reduce the risk of misinterpretation of the search-result. It might be clearer to the user what is what, but at the same time it makes use of more space. Therefore it will result in more row-breaks.

¹ “Paul Smith” counts as two key-presses

8.4 Personal Number

8.4.1 Select Profile

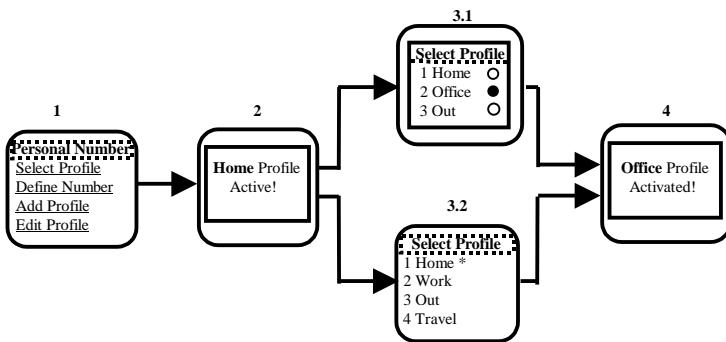


Figure 8.8. Select Profile menu, (two alternatives)

General idea(s)

When selecting the “Select Profile”-option a popup-window (see WML reference) will appear showing the active profile, (fig. 8.8:2). If the number of profiles is greater than the number possible to show on the display, the active profile might not be visible. To handle this situation we created the popup-window. Then a card is displayed where the user selects a profile to set as active, (fig. 8.8:3.1 and fig. 8.8:3.2). After selection of profile another popup-window is displayed confirming a new profile has been selected, (fig. 8.8:4).

In one alternative we decided to use radio-buttons to clearly show the active profile, (fig. 8.8:3.1). This also gives the user an opportunity to change his mind and choose a new profile before leaving this service. One disadvantage with this proposal is that one additional key-press is necessary to choose and confirm the selection. This can be confusing for a user who is unfamiliar with radio-buttons and does not know how to make a selection. This seems to stay in opposition of what we previously stated about the users pre-knowledge and know-how of existing interfaces. However, in the scenario discussed, the user *does not have a mouse/computer keyboard to make the selection, but the keypad of the phone. As the radio-button is not an element of a mobile-phone interface, users do not know how to operate it.* Instead of using the key that is specified for marking a radio-button the user might press the “YES”-key, which actually terminates the “Select Profile”-card and leaves the active profile unchanged. In short: The user fails.

The other alternative, (fig. 8.8:3.2), is designed to minimize the number of key-presses needed. In this design the user selects profile by pressing “YES” when the desired profile is highlighted. The currently active profile is marked with an “*”. To change profile should be easy since only one key-press is needed. This alternative should minimize the risk of mistakes because the same key is used for all selections.

One disadvantage is that the user can not undo the selection unless entering the service once again. The use of a “*” (fig. 8.9) to show active profile might both be confusing for a new user – as this is not a frequently used method – as well as being confused with the “star”-symbol used in telephone numbers or services.

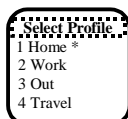


Figure 8.9 Use of “*” in menus.

8.4.2 Define Number

Add Number

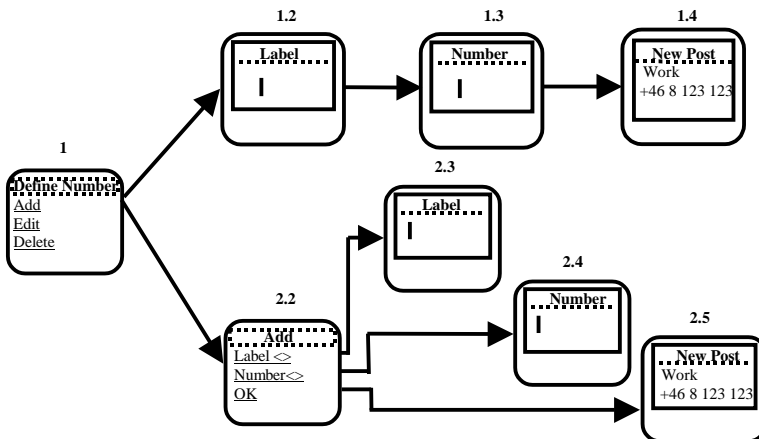


Figure 8.10. Add Number, (two alternatives)

The design in fig. 8.10:1.2-4 was developed to get quick input with few key-presses. After the new number is defined a popup-window informs that a new number has been created, (fig. 8.10:1.4). It should be a fast method since few key-presses are needed. It also should suit a frequent user who is familiar with the PSCS and knows what is needed to specify the new number. This alternative gives the user an opportunity to go back to the previous card by pressing the “NO”-key to make changes. One negative aspect is that the user does not get an overall picture of how many input-fields have to be filled in.

The design in fig. 8.10:2.2-5 was developed to give the user information of the needed input for the new number, and also to show the user what he has entered. It is possible to make changes by reselecting an option. This alternative demands more key-presses then the prior, which may cause irritation for a frequent user who knows what this service offers.

Edit Number

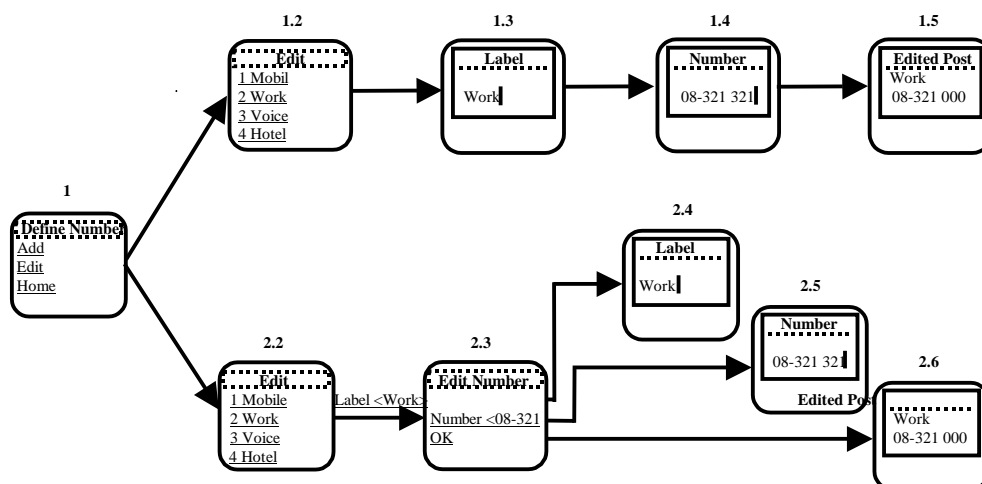


Figure 8.11. Edit Number menu, (two alternatives)

These two alternatives have the same differences as those described in “Add Number”, (part 8.4.2.1). The difference is that the user first has to select which number to edit. We created these two alternatives according to the previous menus to achieve consistency in the design of the application.

Delete number

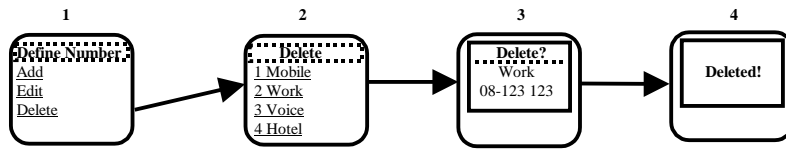


Figure 8.12. Delete Number menu

For this menu only one design has been developed. The user first has to choose a number to delete, and is then asked to confirm the choice to prevent deleting the wrong number.

8.4.3 Add Profile

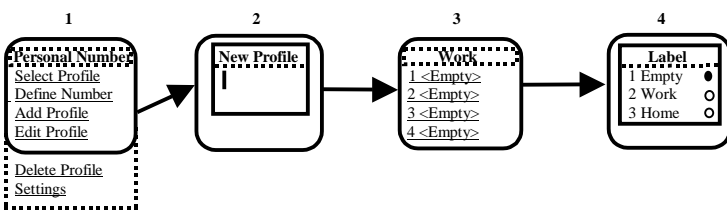


Figure 8.13. Add Profile menu

This is a simple and straightforward design. The user first has to specify a name for the new profile and then selects numbers to be included in the profile. The default value for the numbers in fig. 8.13:3 is “Empty”, which means that no number is selected. Not all number have to be specified. The numbers can be selected from a list containing the numbers defined in “Add Number”, (part 8.4.2.1).

8.4.4 Edit Profile

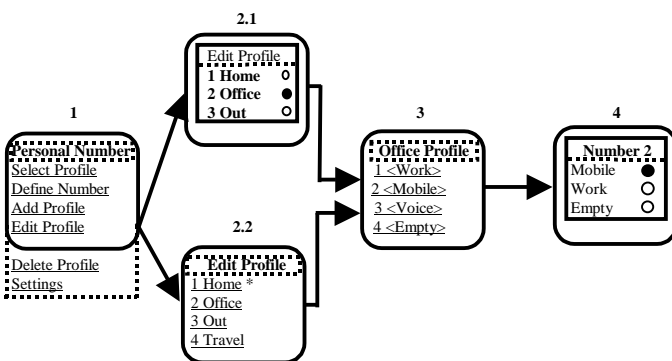


Figure 8.14. Edit Profile menu, (two alternatives)

This design is based on the one for “Add Profile”, (part 8.4.2.1). The difference is that the user instead of creating a new profile selects an existing profile to edit. The discussion about the design of fig. 8.14:2.1 vs. the one in fig. 8.14:2.2 is the same as in “Select Profile”, (part 8.4.1.1).

8.4.5 Delete Profile

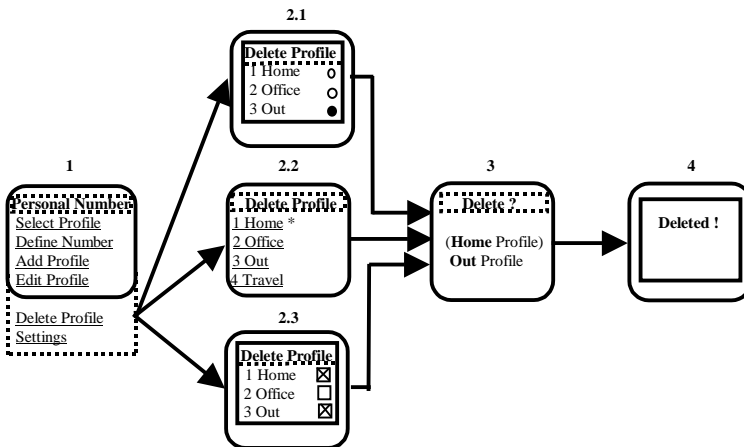


Figure 8.15. Delete Profile menu, (three alternatives)

The difference between these three alternatives is the way the user selects which profile(s) to delete. For the first two alternatives, (fig. 8.15:2.1 and fig. 8.15:2.2), the discussion is the same as in “Select Profile”, (part 8.4.1.1). The third alternative, (fig. 8.15:2.3), would give the user the opportunity to delete several profiles simultaneously. If alternative three is used, all the selected profiles will be listed in the display, (fig. 8.15:3), and at the same time the user is asked to confirm the choice to prevent deleting the wrong number.

8.4.6 Settings

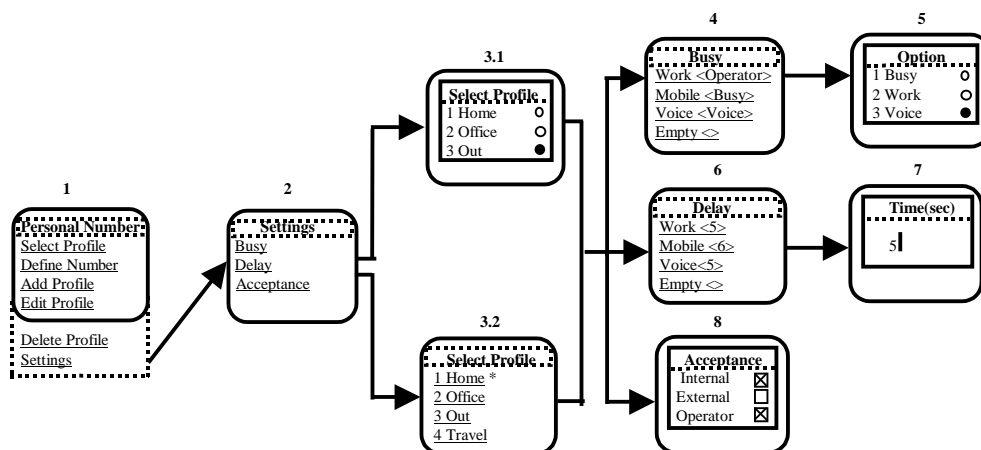


Figure 8.16. Settings menu, (two alternatives)

Use of this menu enables the user to change different settings for a profile. The selection of profile is done according to the design of fig. 8.16:3.1 or fig. 8.16:3.2. The discussion about these two alternatives is the same as in “Select Profile”, (part 8.4.1.1). We have placed the “Settings”-card before the “Select Profile”-card in order to let the user first choose a setting and then select a profiles to apply this setting on, rather than the other way around. We think it is more common that a user wants to change a setting on several different profiles, rather than change different settings on one profile.

The chapters below explain some frequently used ways to handle incoming calls for this service.

Busy

The “Busy”-option is used to specify what to do when the current number is busy, (fig. 8.16:4). The user can here specify if the call should be diverted to another number or signaled as “busy” tone.

Delay

The “Delay”-option is used to specify for how long a certain number should be paged before further diversion. The user first selects which number to edit, (fig. 8.16:6), then an input-field is shown where the actual delay value is specified, (fig. 8.16:7).

Acceptance

The “Acceptance”-option is used to specify what sort of calls are to be indicated. Ticking the different options shown in fig. 8.16:8 makes the selection. This way the user can accept or reject e.g. internal, external, or operator calls.

9 Second Userstudy Iteration

9.1 Testpanel

The testpanel we used in our introductory userstudy was also used in the second one. We also let two persons evaluate the interface together so that they could exchange thoughts with one another, and this seemed to be a good idea since many useful proposals were recorded during this test.

9.2 Purpose

The purpose of this userstudy was to eliminate the designs that were not appreciated by the users and to improve those, which seemed to be most useful. When doing the simulation we asked the user to perform certain assignments related to situations that might turn up in real-life. The tasks contained finding information regarding a fictitious person using the “Search”-service, define new numbers and add them to their phone book, create a new profile and edit it etc, (see appendix A for a complete list of tasks). The users were asked to think aloud while browsing through the different menus. Those thoughts were documented by us to be used in the following redesign. We also noticed which keys the user pressed while performing different tasks and this gave quite a clear picture of how the user acts in different situations. After every subpart of the study we asked the testpanel which of the different proposals they found most useful and if there were any doubts while using the application. They also got a chance to come up with their own thoughts and proposals for improvement.

9.3 Implementation

Based upon our first userstudy (chp. 8) and our explorative sketching we created a paper “mock-up”¹⁶. The different designs we came up with were drawn as a kind of storyboard that served as the basis for our userstudy. During the userstudy the telephone was simulated with a paper model and we presented all the different cards to the users as they were browsing through the menus, (fig. 9.1a).



Figure 9.1a. Userstudy with paper “mock-up”.

¹⁶ Shneiderman, Ben, Chapter 4.3

It had been preferable to show the menu system on the Ericsson WapIDE, (fig. 9.1b), a development kit for WAP-services, but it appeared to be insufficient. The WapIDE includes a microbrowser with an interface like the Ericsson R320, in which you can develop your own WAP-applications using WML.



Figure 9.1b. Ericsson WapIDE SDK¹⁷

Our contacts at Ericsson worked with an older version of the WapIDE than the one available for downloading on the Internet. We did our application test on the new version and since Ericsson had updated the new WapIDE, some functions were changed. For example the way text input is made and single or multiple selection with radiobuttons or checkboxes, (fig. 9.2).



Figure 9.2. Text input, “Radiobuttons” and “Checkboxes”¹⁸

To make use of the benefits of each application, we took the best part from each of the two WapIDE versions and produced an interface, which we thought was both clear and easy to use. The test applications that we made on the computer with the new version of WapIDE showed that when you implement a menu according to the Ericsson style-guide you get an interface that is both complicated and contains unnecessary steps, (fig. 9.3).

¹⁷ Ericsson WapIDE SDK User Guide (991203)

¹⁸ Ericsson Design Guidelines for R320



Figure 9.3. Pictures showing the extra step (middle card) when making a hyperlink (Add) to an input-field.

Due the lack of options we decided to develop our own interface as a paper “mock-up” where every menu follows the Ericsson guidelines. The WapIDE does not yet support some of those functions that we assume here. One of the functions we assumed is the possibility of executing the actual search directly from the input-field by pressing the “YES”-key instead of jumping back to the previous menu, (fig. 9.4).

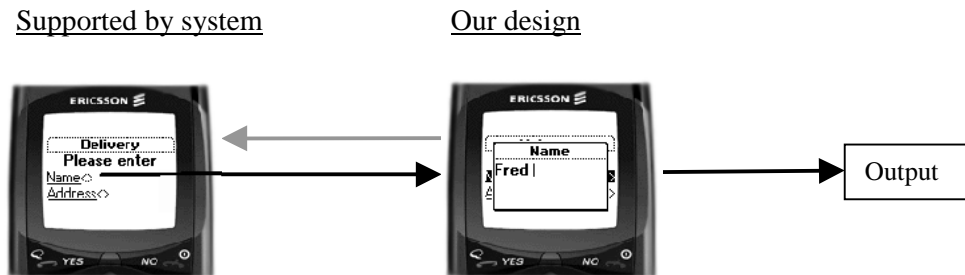


Figure 9.4. Supported by the system vs. our design

This can be seen as a sort of "functionality/interaction overloading", i.e. the context will determine the actual operation to be performed.

Compare this example with two of the search alternatives we created, where alternative 1, (part 8.3.2), shows a function supported by the old WapIDE and alternative 2 a function supported by the new version, (part 8.3.3). When combining these two versions we give Ericsson different proposals to work with.

9.4 Factors that may have influenced the testpanel

One of the benefits with a paper “mock-up” is that it is easy and flexible to change the interface when re-designing. On the other hand the user is not given the same opportunity to explore the interface by “trial and error”. Real prototypes with the keys and a display give users the opportunity to try and figure out how it really works. Our paper “mock-up” demands the user to press the correct keys, since it does not give any error messages and it is impossible to predict all possible scenarios that might appear. Another disadvantage with a papermodel is the way the testpanel experiences the interface. It is common that they do not treat the model like a real telephone, but just say which option they wish to choose instead of using the “arrow”-keys or the “YES” or “NO”-keys. It is then hard to get a clear picture of how strenuous it can be to browse between different options. It is the dynamics of an application that are missed out upon.

There are also other factors that might have affected the testpanel during this trial iteration. Our menu headlines were not always informative and tended to mislead the users. At some occasions the study was interrupted by phone calls the users had to answer, which may have distracted them. Some of the users were also in a hurry and did not have time to attend to the whole userstudy. Also, one of them had been involved in the development of the “Personal Number”-service, and – not surprisingly – found the system to be quite satisfactory.

9.5 Evaluation and redesign

9.5.1 Search (Directory)

Alternative 1

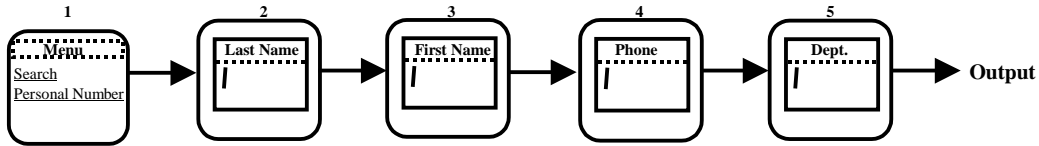


Figure 9.5. Search menu before redesign, (alternative 1)

Response

This design mostly got negative responses from our testpanel. Only one of the users preferred this alternative among the ones presented. The majority of the testpanel got confused when all the input-fields showed up, (fig. 9.5:2-5). Most of the users expected the search-result to show up in the display when pressing the “YES”-key after having filled in the input-fields for last and first name. Instead the input field for phone number appeared. This would not be a problem if only one input field had to be filled in, but since there are different search-topics the application can not predict when to execute the search. The userstudy showed that the users preferred to know how many input-fields to fill in.

In spite of the fact that the majority of the group did not see any benefits with this method, we think that a user familiar with this way of interaction would appreciate this direct way of searching for a person. We also think this would be the fastest method. Still, when mentioning this for the participants in the userstudy, no one changed their mind.

The problem with how to show the user which button to press when trying to get to the next card became obvious when one of the participants pressed the “NO”-key instead off the “YES”-key.

Suggestions left unused

One wish brought up was an AutoFill-function when typing the names. Instead of the user typing the whole name, the application helps with some suggestions when typing the first letters. While we agree that this would be a nice feature, we also fear that this might require a lot of data to be transferred between the terminal and the WAP-server, making this feature unacceptable slow. The idea is therefore left for feasibility investigation.

Redesign

Due to the lack of response for this alternative, we have decided not exclude this design option from further iterative improvement.

Alternative 2

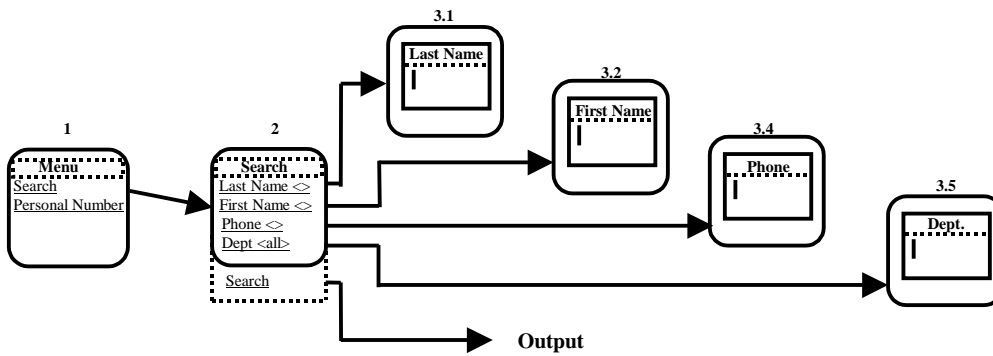


Figure 9.6. Search menu before redesign, (alternative 2)

Positive response

In this proposal the testpanel appreciated the feedback they got from the application at all times. Right from the start it is obvious how many input-fields have to be filled in, since they are all visible. It is easy to decide which search-criteria to enter and the user will not be “forced” into any unwanted input-field. This is also the major difference between this design and the previous one, (part 9.5.1.1).

Negative response

When the user had filled in the input-fields he/she wanted, there was some confusion how to execute the actual search. Most of the users instinctively pressed the “YES”-key, which resulted in returning to the last input-field visited. The problem here was the limitation of the display size. We could not place all five options on one card. As a consequence the selection of the “Search”-option required the user to scroll down in the GUI. The Ericsson software unfortunately does not support an indicator showing that there is more information below like the one by Nokia, (fig. 9.7).

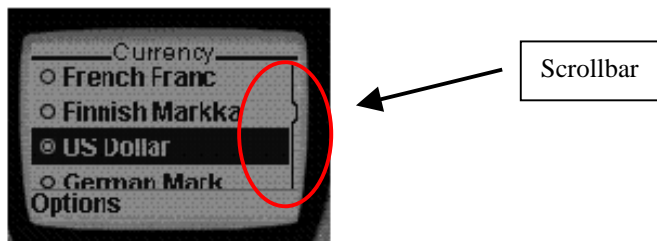


Figure 9.7. Scrollbar by Nokia¹⁹

¹⁹ Developer’s Guide NOKIA WAP TOOL KIT Version 1.2
http://www.forum.nokia.com/wap_developer/download/sdevguide.pdf (991204)

Suggestions left unused

One suggestion was to introduce an option called “More” or “Down” and situate it at the bottom of the display. This was meant to inform the user that more alternatives are available below. Another suggestion was to create a menu that only shows one option at a time. The user would then have to scroll through the options connected in a loop. These two ideas were left unused because a, what we believe is a better alternative was brought up, which we used in the redesign (see below).

Redesign

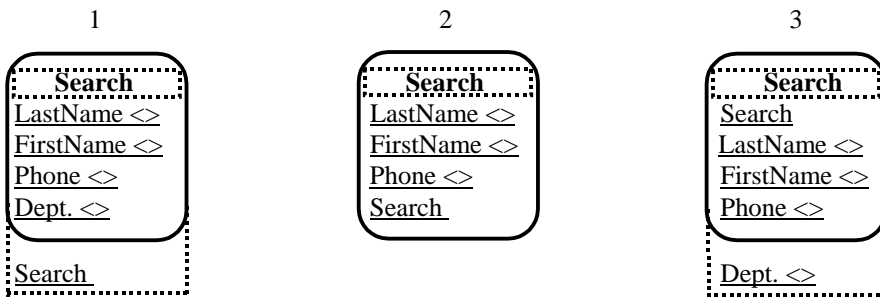


Figure 9.8. “Search”-card, (three alternatives)

When working on the design for the “Search”-card, (fig. 9.9:2), we had some different suggestions to choose between. The one that was used in the previous design, (fig. 9.8:1), raised some questions from our users of how to proceed after the search criteria wanted had been entered.

One suggestion we encountered through the userstudy was to remove the option for entering department as search criteria. This way the “Search”-option would be visible when entering the “Search”-card, (fig. 9.8:2). Among the users we have studied, the common opinion was that the “Department”-option could be removed, because they believed they would not use it. Once again the question is raised: Is our testpanel representative enough for the target users? They are all employed inside Ericsson and the department-names at Ericsson are somewhat complicated. But what about other companies? Maybe other users would benefit from this option. This question will not be answered unless userstudies are performed on people outside Ericsson. Lacking the possibility to do an increased user-study at this stage we have decided to leave this option.

Another suggestion encountered was to simply place the “Search”-option further up and instead hide the “Department”-option, (fig. 9.8:3). This way the “Search”-option would be visible from the start, and the “Department”-option still there for those who wants to use it. One thing that could be a problem is that the user might press “YES” right away when the “Search”-option is highlighted. To solve this problem we simply made the “Search”-option unselectable until some search criteria is entered.

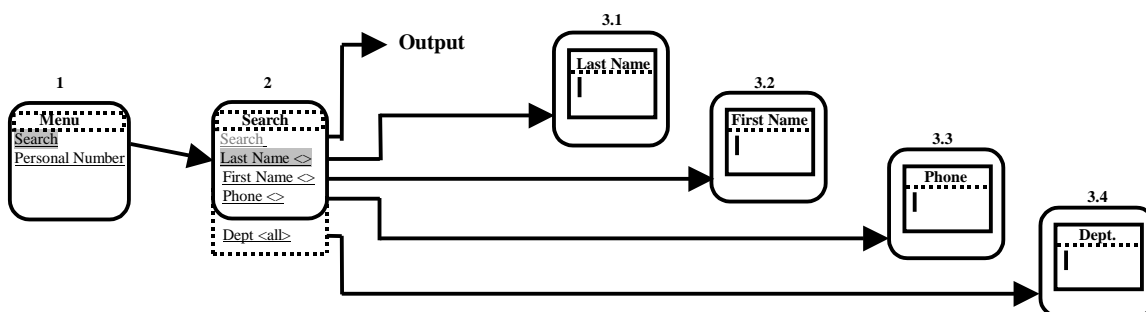


Figure 9.9. Search menu after redesign, (alternative 2)

Alternative 3

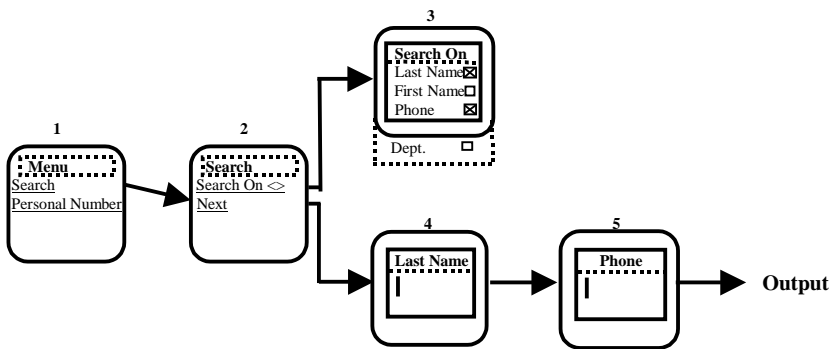


Figure 9.10. Search menu before redesign, (alternative 3).

Response

This alternative showed to be least appreciated among the ones presented. The testpanel got confused from the start when the first card was shown in the display, (fig. 9.10:3.2). The name "Search On" was not informative enough and most of the users did not understand how to proceed. When they finally pressed the "SearchOn"-option and got to the "Search On"-card with the four checkboxes,(fig. 9.10:3), they still did not know what to do. Nor was the name "Next" a good choice. As a result two of the users reflected that they were now done and could move on to the next task after selecting this option. The general opinion was that the design was too lengthy to work with.

Redesign

Although this alternative did not get too much positive response, we have decided to make a redesign. This because we think the new design, (fig. 9.11), will be much more appreciated by the testpanel. In the new design the user is presented the different search-topics right away. The users can then simply tick the topics they want to use as search-criteria. When leaving this card by pressing the "YES"-key, the user is directly presented input-fields for the different topics chosen, (fig. 9.11:3-4). To leave the card at least one topic has to be ticked, (fig. 9.11:2), otherwise pressing the "YES"-key will give no response.

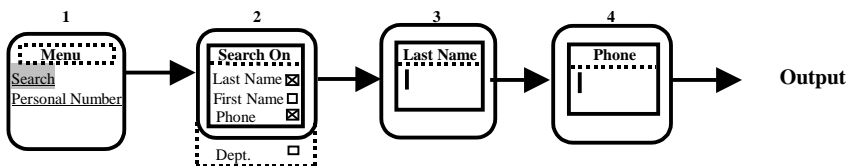


Figure 9.11. Search menu after redesign.

Output

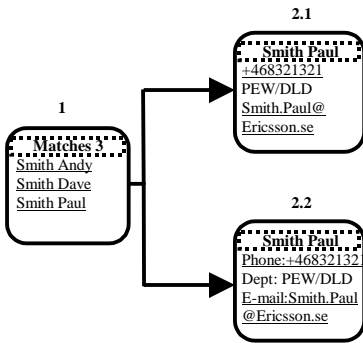


Figure 9.12. Output before redesign, (two alternatives)

Redesign

The choice here was whether to write ‘phone’, ‘department’ (‘dept’) and ‘e-mail’, in front of the search-result or not, (fig. 9.12:2.1 and fig. 9.12:2.2). Most of the users did not notice the difference. When it was pointed out, most users preferred the alternative without headings. This alternative also gives fewer abbreviations.

Without the extra information one problem could be to recognize what is what, although the testpanel did not refer to this as a problem. However, one of the users mentioned that this problem might be more significant outside Ericsson, where companies perhaps has departments named with digits instead of letters. It then might be harder to separate phone number and department. To solve this problem we changed the order in which the result is displayed by putting e-mail between phone-number and department, (fig. 9.13:2). We count on e-mail still being intuitive due to its distinct format, including the ‘@’ symbol.

The possibility of making a phonecall or sending an e-mail only by pressing the “YES”-key when respective output is highlighted was very appreciated.

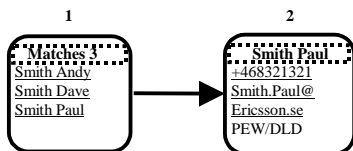


Figure 9.13. Output after redesign.

9.5.2 Personal Number

Select Profile

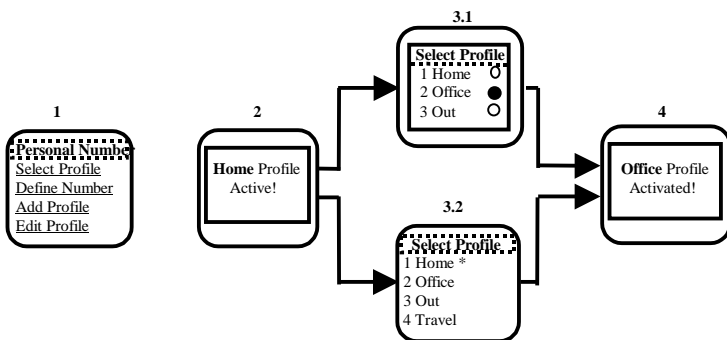


Figure 9.14. Select Profile menu before redesign, (two alternatives).

Redesign

In this case the main question was whether to use “radiobuttons”, (fig. 9.14:3.1), or the alternative shown in fig. 9.14:3.2, when the user has to select a profile. The study did not give us a unanimous answer to this question, but those who had an opinion voted for the second one. We also noticed that several of the users repeatedly pressed the “YES”-key to select a choice instead of the “Right arrow”-key. With this in mind we went for alternative 2. The “*” displayed indicates which profile currently is active.

The popup-windows are shown during a limited time-interval, (fig. 9.15:2 and fig. 9.15:4), but the user is also given the opportunity to clear them by simply pressing the “YES”-key.

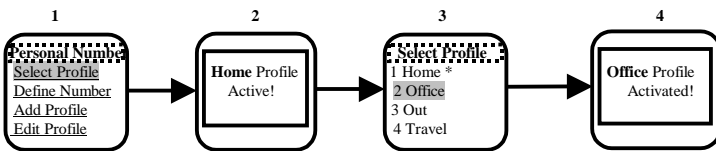


Figure 9.15. Select Profile menu after redesign

Define number

Add number

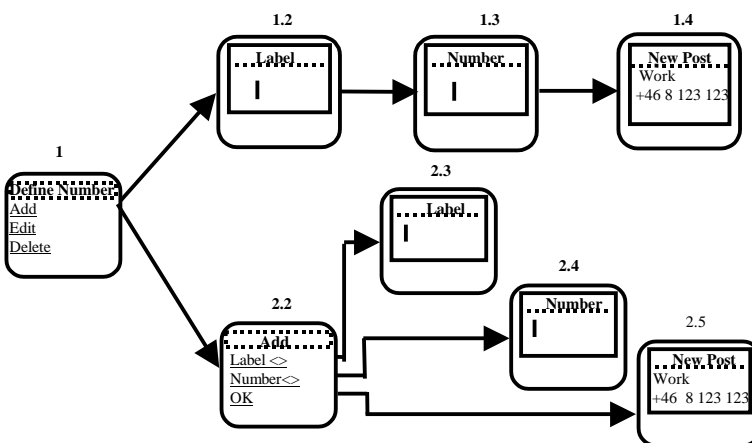


Figure 9.16. Add Number menu before redesign, (two alternatives).

Redesign

Here we choose between the alternatives shown in fig. 9.16:1.2-4 and fig. 9.16:2.2-5. The userstudy did not show that one design was better than the other, but in order to achieve a consistent design throughout the structure we decided to follow two simple rules.

When adding a number the user has to fill in input-fields for both “Label” and “Number”, (fig. 9.16:1.2). In this case the input-fields appear in a linear sequence independent of the users choice²⁰ to prevent the user to leave out significant input information needed for the task to be completed successfully. We are aware that this required stepping violates the usability guideline of the user having control over the application at all times. We believe this to be justifiable here.

Our second ”rule” deals with editing an existing number. Here the alternative according to fig. 9.18:2.3-6 below give the user the choice to edit either “Label” or “Number”. We have used this design to prevent

²⁰ Shneiderman, Ben, Chapter 7.2

the user from being “forced” into an input-field that does not have to be edited. These two ways were actually suggested by one of the users during the userstudy.

The design of the “popup window” is the same as in “Select Profile”, (part 9.5.2.1).

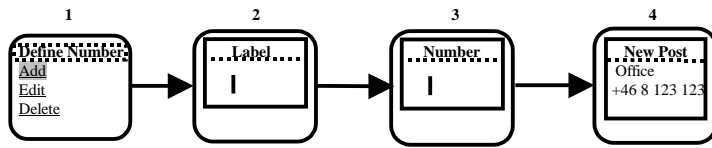


Figure 9.17. Add Number menu after redesign

Edit number

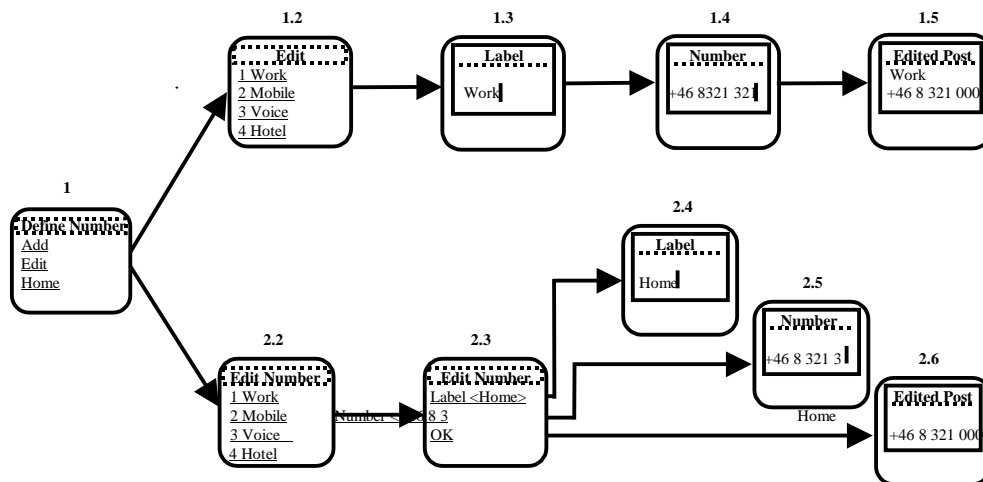


Figure 9.18. Edit Number menu before redesign, (two alternatives)

Redesign

Regarding the “Edit Number”-card, (fig. 9.19:2), the design is based on the same discussion as in part 9.5.2.1 “Select Profile”, the same goes for the popup-window, (fig. 9.19:4.3). For the “Edit Number”-card the design is based on the discussion in part 9.5.2.2.1 “Add Number”.

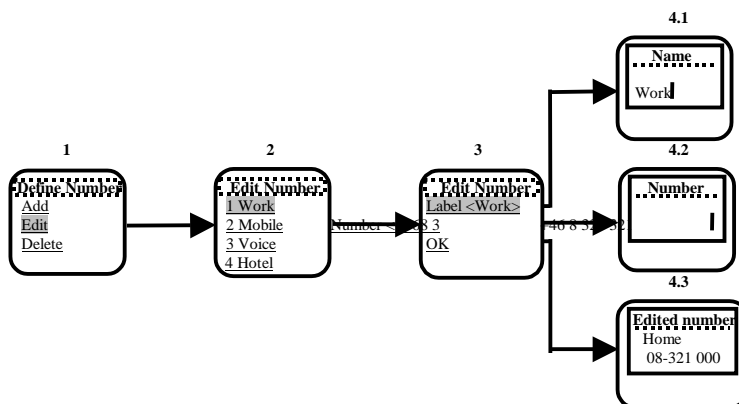


Figure 9.19. Edit Number menu after redesign

Delete number

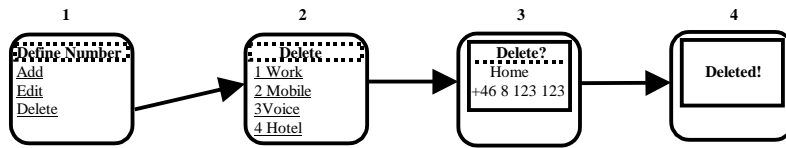


Figure 9.20. Delete Number menu before redesign

Redesign

To prevent users from deleting a number by mistake we have redesigned the “Delete?”-card, (fig. 9.21:3). After entering this card the user must press the “Arrow down”-key and then the “YES”-key in order to confirm the delete-choice.

We did so because we imagined the following scenario: If the system is waiting for response from the server the user might experience this as if “nothing is happening” and therefore he/she might press the “YES”-key repeatedly. If the “Delete”-option in the “Define Number”-card is highlighted, (fig. 9.20:1), this might delete a number by mistake. With the redesign of the “Delete?”-card this will not happen, (fig. 9.21:3).

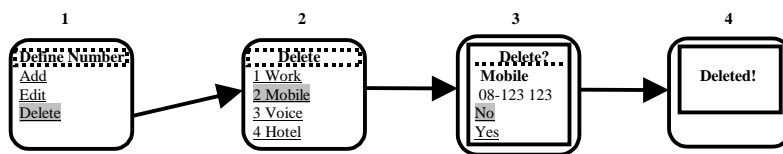


Figure 9.21. Delete Number menu after redesign

Add Profile

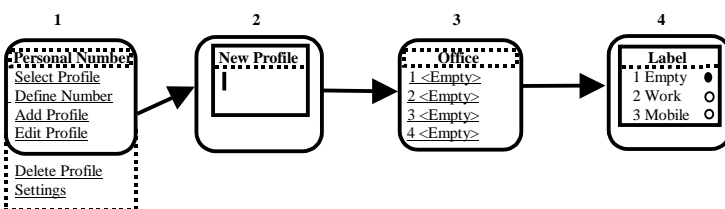


Figure 9.22. Add Profile menu before redesign

Redesign

The last study showed that the names “New Profile” on the card, (fig. 9.22:2), and “Label” on the card, (fig. 9.22:4), was not informative enough for the users. We changed these headings to “Enter name” and “Number” respectively.

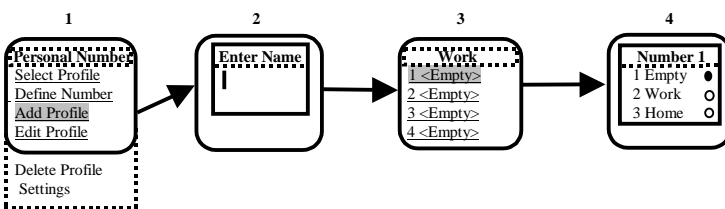


Figure 9.23. Add Profile menu after redesign

Edit Profile

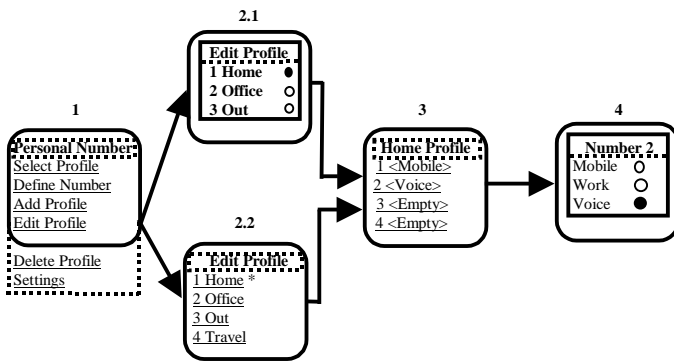


Figure 9.24. Edit Profile before redesign, (two alternatives)

Redesign

Regarding the “Edit Profile”-card, (fig. 9.25:2), the design is based on the same conclusions as in part 9.5.2.1 “Select Profile”. We also discussed whether or not to display the “*” indicating which profile is currently active. In the “Select Profile” case this is relevant information for the user, (part 9.5.2.1), but is it necessary here? Will it not perhaps confuse the user? We could not get a clear opinion about this and therefore regard this issue as not clearly answered. We will therefore stay with the current implementation.

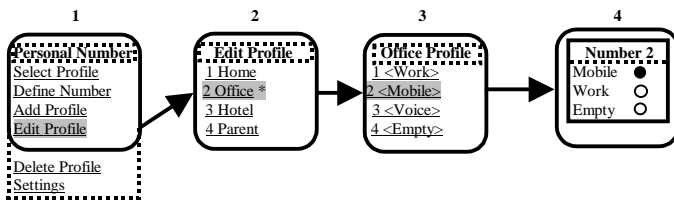


Figure 9.25. Edit Profile after redesign

Delete Profile

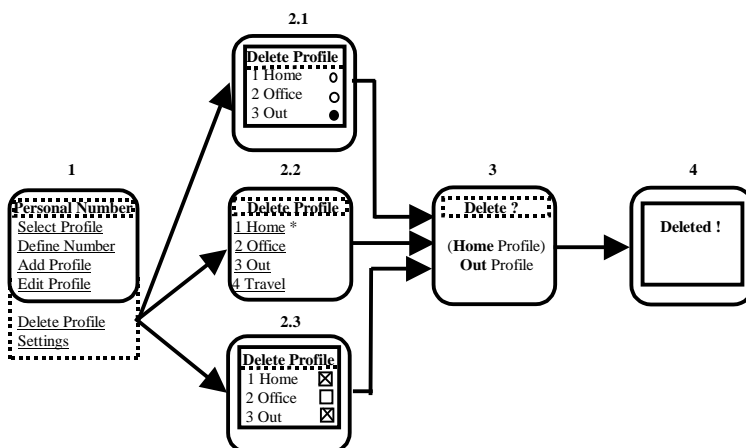


Figure 9.26. Delete Profile menu before redesign, (three alternatives)

Redesign

One of the testpanel found the multiple “Delete Profile”-card to be useful, (fig. 9.26:2.3), but the question is: How many profiles does the user really create? We think this option is unnecessary since the user will probably only create a few profiles and deleting them will be a even more rare used function.

Regarding the “Delete Profile”-card, (fig. 9.27:2), the design is based on the same conclusions as in part 9.5.2.1 “Select Profile”, the same goes for the popup-window, (fig. 9.27:4). For the “Delete?”-card, (fig. 8.27:3), the design is based on the discussion in part 9.5.2.2.5 “Delete Number”. About the question whether or not to display the “*” in the “Delete Profile”-card, (fig. 9.27:2), indicating which profile that is currently active, the discussion is the same as in part 9.5.2.1 “Edit Profile”.

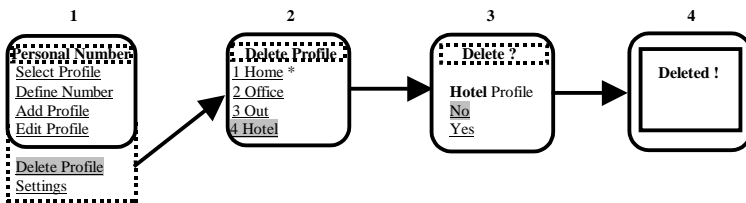


Figure 9.27. Delete Profile menu after redesign

Settings

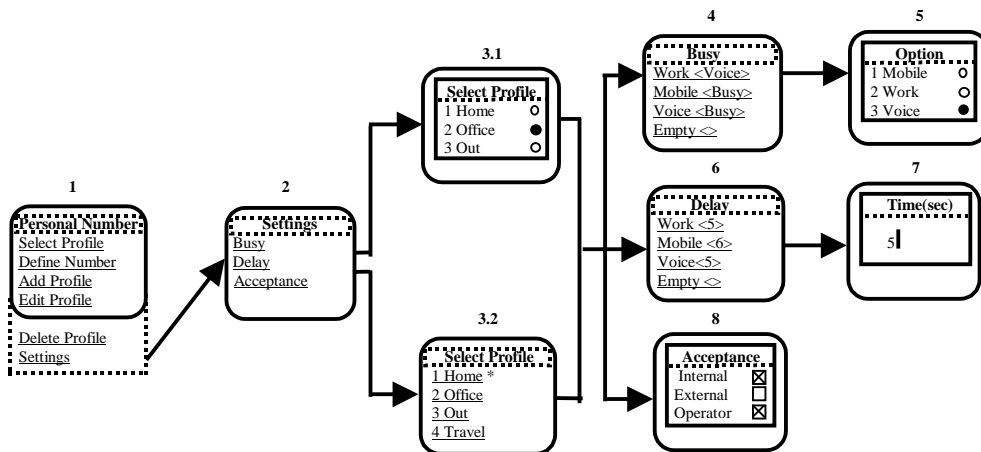


Figure 9.28. Settings menu before redesign

Redesign

In the “Time”-card the user has the possibility to increase and decrease the delay-time by using the “Arrow”-keys, as well as just entering a number.

The design of the “Select profile”-card and the “Setting”-card, (fig. 9.29:2 and fig. 9.29:3), is based on the same conclusions as in part 9.5.2.1, “Select Profile”.

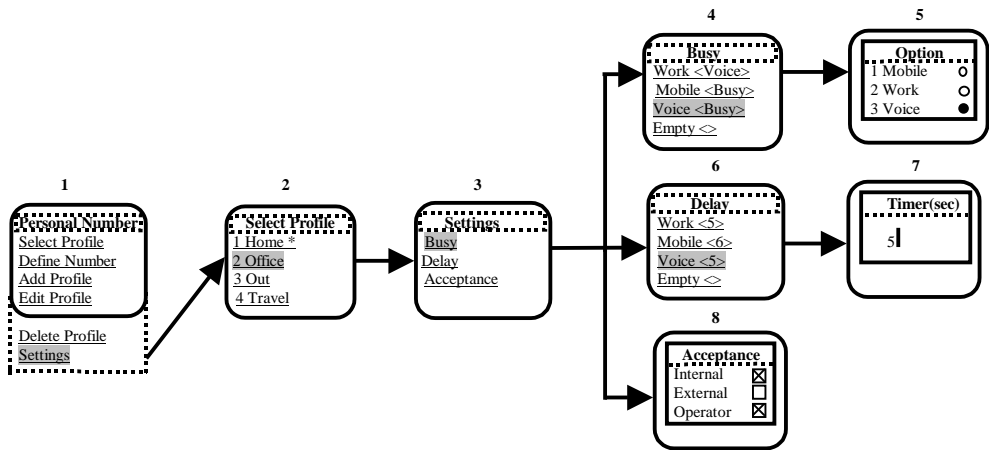


Figure 9.29. Settings menu after redesign

10 Third Userstudy Iteration

10.1 About the userstudy

This third userstudy iteration was performed in the same way as the second one. The difference was that we now had fewer proposals after having evaluated the second userstudy. This resulted in a more detailed study of the different alternatives. We did not get to interview all of the participants from the first userstudy since some of them were unavailable. Thus, we had to get in contact with some new users at Ericsson and this gave us some new thoughts, which had not been brought up earlier.

10.2 Evaluation and redesign

Of the new ideas that were brought up during the last userstudy, we used the ones we thought would improve the design to a great extent. The design ideas that at this point still needed further iterative design improvements were not considered any more.

One thought that concerned both “Define Number” and the “Profile”-options was the suggestion to design the interface more like a Microsoft Windows application. In a Windows kind of application, one first selects an item (here for example a Profile) and then decide what to perform on it, e.g. coming to a screen where it can be edited or deleted. This is a conceivable idea, which should be investigated further.

10.2.1 Search (Directory)

Alternative 1:

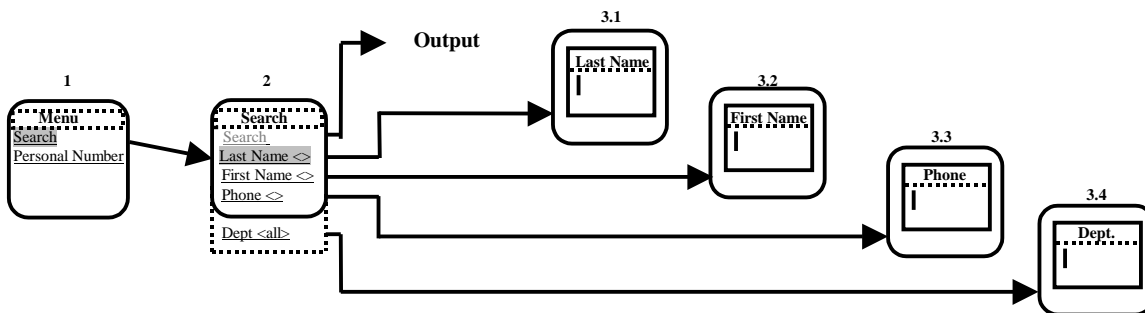


Figure 10.1. Search menu, final design.

Response

Several of the users tried to write the name directly between the brackets instead of first selecting the hyperlink to get to an input-field. We think this is a good suggestion, but it is, to our knowledge, not yet supported.

The disadvantage with this approach is lack of space on the display. The application will simply divide the name into smaller parts and possibly hide parts of it, (fig. 10.2).

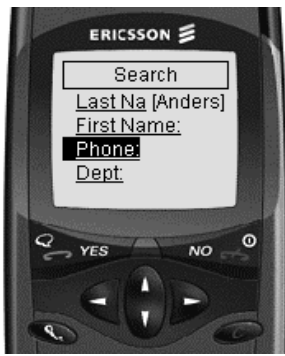


Figure 10.2. “LastName” specified as “Andersson”

One idea was to execute the actual search directly from the input-field. If the search results in too many matches the user can press the “NO”-key to get back to the “Search”-card and specify more search-topics. This way however, excludes the possibility of specifying e.g. both last and first-name before executing the search, (fig. 10.3).

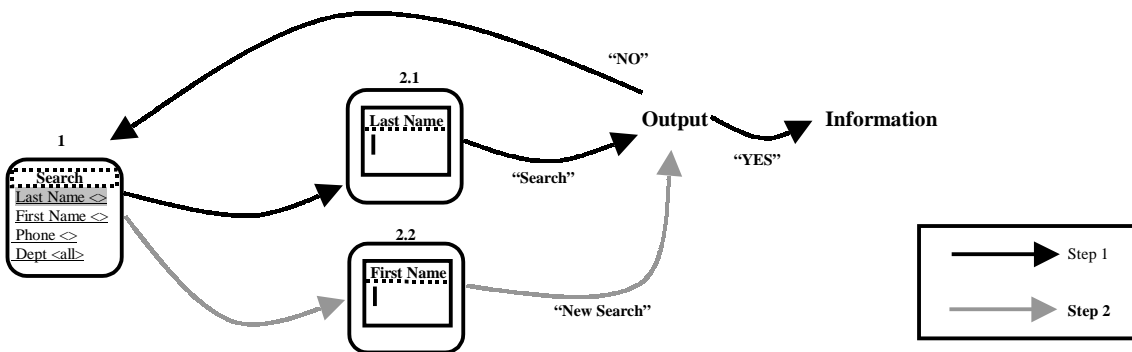


Figure 10.3. Search for both last and first name made in two steps

The alternative 1 was still the most popular. Although the last userstudy gave us some new ideas we decided to keep this design as our final one for this function, (fig. 10.1).

Alternative 2

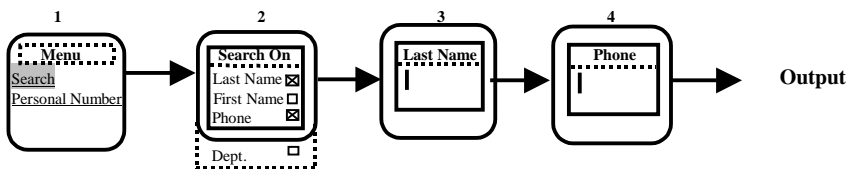


Figure 10.4. Search menu, (alternative 2).

Response

Although this was an improved design of the one in part 9.5.1.3 no one really seemed to like it. We believe now that users do not want to pre-specify their search-topics. Due to the observed user reaction we decided to select the other alternative as our final one.

Output

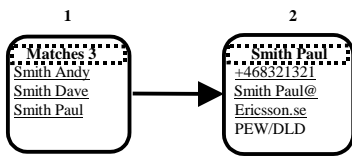


Figure 10.5. Output menu, final design.

Response

This design got positive response in the userstudy. Although we got a suggestion that the user should be able to dial a number directly from the “Matches”-card, (fig. 10.5:1). As it is designed right now, the user first has to enter the information-card by pressing the “YES”-key, (fig. 10.5:2), and then press the “YES”-key once again. If we could implement different options for different keys, the user could be given an additional option, to dial directly from the “Matches”-card, e.g. by pressing the “Right arrow”-key, (fig. 10.6).

As this would not be consistent with the Ericsson keypad-action model as of today, we foresee considerable opposition to such a design idea.

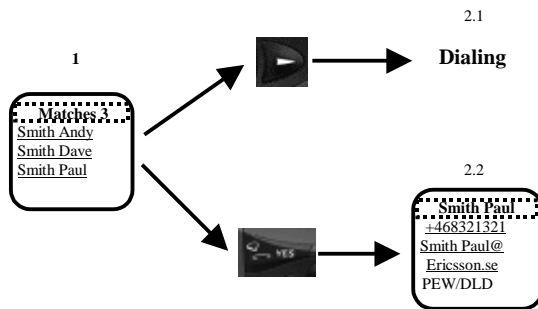


Figure 10.6.

10.2.2 Personal Number

The problem encountered earlier still prevailed. The user did not know which option to choose in order to edit or delete a number. The name “Define Number” obviously led the user think that the only option here was to add a number.

The “Personal Number”-card is also inconsistent since it does not contain a name gathering ‘add’, ‘edit’, and ‘delete’-profile under one option like “Define Number”. One solution is to reduce the number of options in the “Personal Number”-card and create separate ones for “Number” and “Profile”. These two options should then contain necessary submenus. Replacing “Define Number” with “Number” and placing “Add”, “Edit”, “Delete”-profile and “Settings” under “Profile” also reduces the risk of choosing the wrong option, (fig. 10.7:2-4). Because the “Select Profile”-option seems to be frequently used we saved this as a separate option.

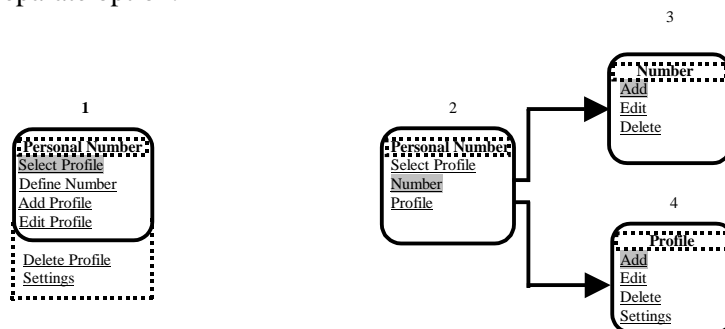


Figure 10.7. Separated Menus for the changing (add, edit, delete, [settings]) for Numbers and Profiles

Select Profile

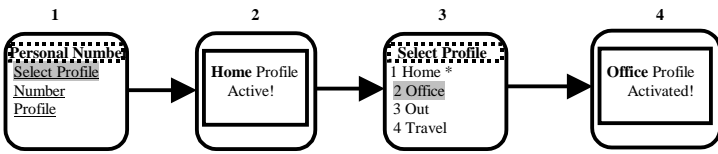


Figure 10.8. Select Profile menu before redesign.

Response

The design we used in this userstudy was the same as in the second, and now we got some additional suggestions. One wish was to be able to choose a profile by simply pressing the number associated with the profile. This could be a fast and conceivable solution that we will try to use in the implementation of the application.

Some users also found the first popup-window unnecessary, (fig. 10.8:2), since we have the “*” indicating an active profile. We decided to take this in consideration into the final design.

The use of the “*” was not obvious to everyone and we also got a suggestion to indicate active profile trough “highlighting” instead. We do not think this idea would work since the highlighted profile will change when scrolling in the “Select Profile”-card.

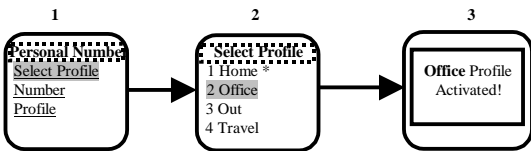


Figure 10.9. Select Profile menu, final design

Define Number

The designs for “Add”, “Edit” and “Delete” in “Define Number” are kept the same since they worked well and got positive response during this userstudy.

Add Profile

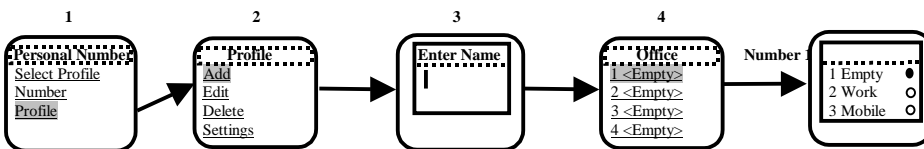


Figure 10.10. Add Profile without confirmation

The userpanel pointed out to us that they missed the option to confirm when being done. This is supposed to be done by pressing the “YES”-key for at least two seconds. This was not obvious so in the final design we have also given the user the opportunity to choose the “OK”-option.

Regarding the “Number”-card, (fig. 10.11:5), the design is based on the same discussion as in part 10.2.2.1, “Select Profile”.

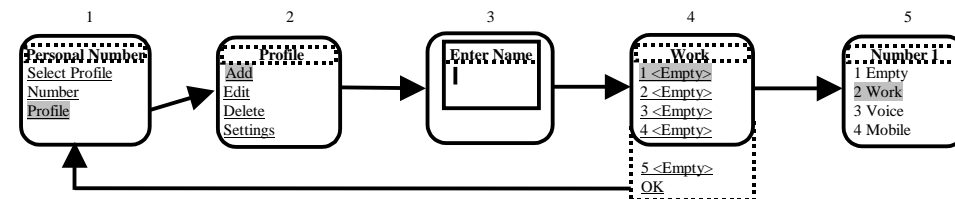


Figure 10.11. Add Profile with confirmation, final version

Edit Profile

The design for “Edit Profile” is kept the same since it worked well and got positive response during this userstudy.

Delete Profile

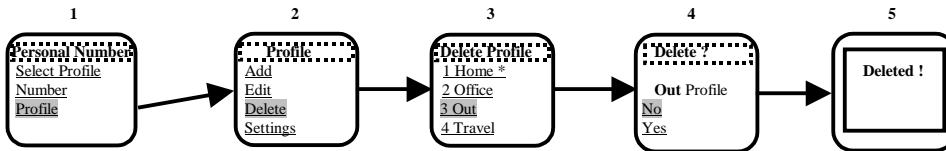


Figure 10.12. Delete Profile with confirmation, active Profile indicated with "*"

The new design of the “Delete?”-card resulted in some confusion, (fig. 10.12:4). When the card appeared some of the users instinctively pressed the “YES”-key to delete. Since the “No”-option was highlighted, nothing was deleted. We assume this will be a minor problem once the user is familiar with the application, as all options are selected by pressing the “YES”-key. Some of the users got confused when they discovered the “*” marking the active profile, (fig. 10.12:3). This because the information is unnecessary for the “Delete Profile”-option. Based on this comment we removed it from our final design.

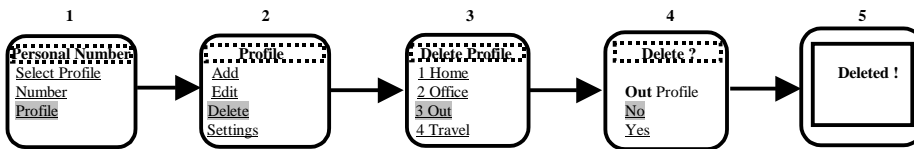


Figure 10.13. Delete Profile with confirmation, active Profile *not indicated*, final design

Settings

The only suggestion we got during this trial upon this function was to make the name of the “Delay”-card more informative to also show the unit of the numbers. We added this information to the final design.

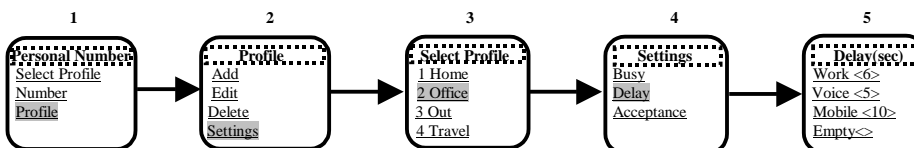


Figure 10.14. Settings with indication of unit used, final design

11 Conclusions

11.1 Reflections

Our most important interface design flow-charts for the selected functions designed as WAP-application are given once more in Appendix F for a result overview²¹. As they actually do not provide much information by themselves, the iterative process producing them, its documentation, and argumentation is to be seen as an important result in itself.

During the project we have gained many new usability experiences in the pioneering field of WAP technology which we believe to be valuable for the development of other WAP-applications as well. The userstudies have shown the importance of cooperating with users when designing a WAP interface. Interaction, designs etc. which we took for granted and thought to be intuitive often showed to be sub-optimal. A good example of this is when the first design of the “Search”-option, (part 9.5.1.1) – that we were almost sure to be among of the most appreciated ones by the testpanel – showed to be a major design miss not appreciated at all.

We also had to be aware of the users having different usage backgrounds. In the beginning of the study we thought that those users owning a GSM mobile phone would manage to perform the different steps of the userstudy without difficulties. However, the only user without access to a mobile GSM phone at all was the one handling our design best. This can be interpreted as our designs not being adapted for the experienced mobile phone users in an adequate way. It can also be based upon the fact that we used the PSCS application as starting point and should perhaps have given greater consideration to the mobile phone instead. Redesigning of an application already released on the market is costly²². Therefore the userstudies are a very important part of the design process, to make sure the users needs are fulfilled. We would consequently like to advise for validating usability studies, possibly taking our design(s) further, where we were forced to leave the further in-depth investigation due to time constrains. This could be part of a feasibility, marketing, or product development supporting activity for example.

Another point, which we would like to stress, is the availability of experts from different fields during the design / usability engineering period. At a few occasions during our userstudies the testpanel felt lost and did not know how to proceed. Mostly because they could not interpret some terms we used in the design. This shows the need to involve experts in different areas during the design-phase of a project like this, e.g. linguists, native speakers, WAP programmers, device engineers etc.

11.2 About the application

The project has shown that selected functions from the PSCS can be designed as WAP-application as far as the user interface is concerned. The final design of the application is at the moment not possible to foresee in all detail, as too many degrees of freedom are not yet specified. If WAP is to be a success, there has to be a further standardization and at least, Ericsson wide co-operation between the different business units involved. If the user of one Ericsson WAP-application can not perform the same tasks when changing to another Ericsson WAP terminal, this will not be a sign of strength for Ericsson. Also the amount and rate of data needs to be carefully monitored. WAP applications of the first generation will need to be fine-tuned for datathroughput to come up to acceptable speeds in the service offering. With the advance of transmission rates, hard- and software progression, WAP technology will be able to move into application areas, which can not be covered with an acceptable level of usability today.

²¹ only an overview is given since the complete scope of all individual interfaces working on the different levels would greatly extend our documentation efforts – as focus is put upon the usability design and not on producing an engineering specification, this seems appropriate.

²² Boivie, Inger, *Användarcentrerad systemutveckling*, 1997

11.3 How to proceed

We would like to propose different possibilities how to possibly proceed. One idea would be to take the further development to the "hardware-level", i.e. continue to develop the application using the existing hardware as starting point. This would ensure that the design developed is compatible with the hardware it is supposed to be used with. The problem is that the design no longer would be based on the users opinions to the same degree. It would instead mostly be based on the limitations of the hardware. The reason we did not include the aspect of the hardware too much in our design, was simply because no hardware was available (we are talking about the Ericsson R320). While the WAP-IDE kit was available, we ran into some troubles with it as explained earlier, (part 9.3). To follow the "hardware" approach one would have to have a hardware to test the application on, and even more important, to use during the userstudies.

Another approach would be to expand the userstudy to include people outside Ericsson, and at the same time increase the number of users studied. This would ensure that the results of the userstudies would be more reliable, and at the same time probably contribute with more suggestions and remarks about the design tested. As WAP is a fast-moving and changing technology, the upcoming changes could be incorporated.

A third thought is to cooperate with a project team developing and implementing the WML-code and the server-application to make the software more compatible with our ideas about the design. Of course a combination of these three options would be the best way to continue and in the end achieve a Ericsson ease-of-use even for the WAP services to come.

12 Appendix

Appendix A

Abbreviations

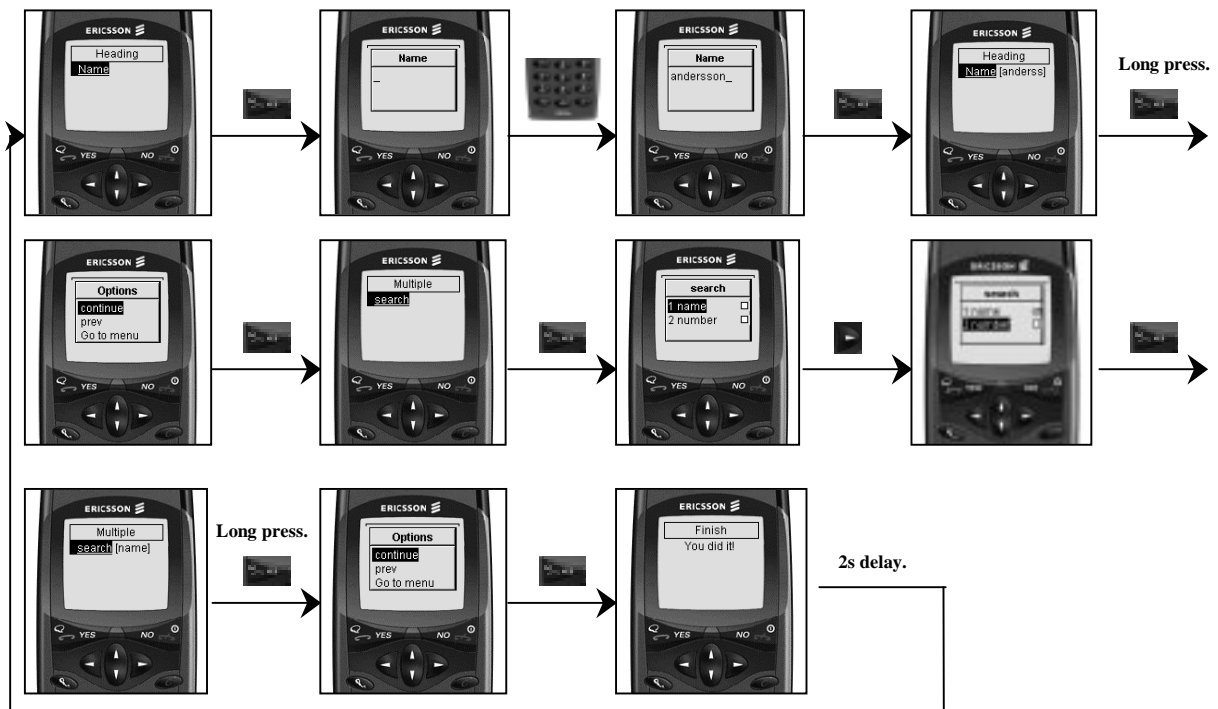
DECT	Digital Enhanced Cordless Telecommunications http://www.ericsson.com/BN/dect.html
GPRS	General Packet Radio Service http://www.whatis.com/gprs.htm http://www.gsmworld.com/technology/gprs.html
GSM	Global System for Mobile communication http://www.whatis.com/gsm.htm
HTML	Hyper Text Markup Language http://www.whatis.com/html.htm
HTTP	Hypertext Transfer Protocol http://www.whatis.com/http.htm .
SIM-card	Subscriber Identification Module http://www.bellsouth.co.nz/network/simcard.htm
SMS	Short Message Service http://www.whatis.com/sms.htm
WAP	Wireless Application Protocol
WML	Wireless Markup Language
WTLS	Wireless Transport Layer Security http://www1.wapforum.org/meetings/developer/

Definitions

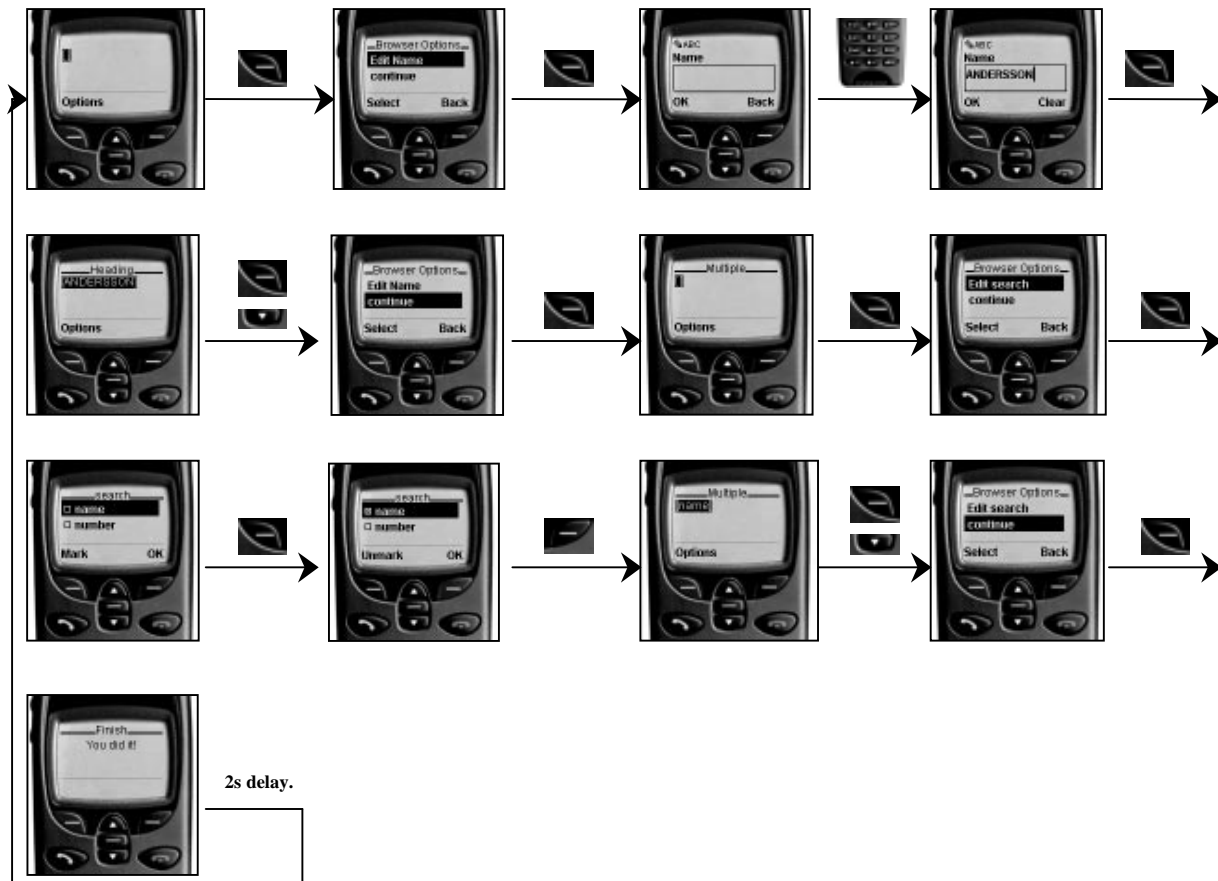
Checkbox	checkboxes are used for the selection of zero, one or more items in a list. Every item in a list has its own Checkbox. When selecting an item the “up-arrow-“ and “down-arrow-button” is used for scrolling and the “right-button” for selection.
Card	single WML unit of navigation and user interface. May contain information to present to the user, instructions for gathering user input, etc.
Card Contents	subject data stored or generated at an origin server. The card contents display text, images, buttons and other controls supported by WML as a response to a user request. When a user performs an action in the card contents, the WAP Browser responds according to the WML specification.
Card Title	An attribute in the WML Card element specifying advisory information about the card.
Current Card	The card whose contents is currently displayed in the browser.
Deck	A collection of WML cards.
Minibrowser	The browser used in a mobile device.
Popup-window	A window that is displayed for a short time and then disappears.
Radiobutton	Radiobuttons are used for selection of one and only one item in a list. For selection of an item the “rightarrow-button” is used. When selecting an item the “up-arrow-“ and “down-arrow-button” is used for scrolling and the “right-button” for selection.

Appendix B

Ericsson WapIDE SDK 2.0B8 (using the Ericsson R320)



Nokia WAP Toolkit 1.2 (using the Nokia 6150)



Appendix C

WLM application code tried

```

<!--Author: David Bezjak --
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
           "http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>
<card id="first" title="Heading">
  <do type="accept" label="continue">
    <go href="#second"/>
  </do>
  <p>
    <input type="text" title="Name" name="name"/>
  </p>
</card>
<card id="second" title="Multiple">
  <do type="accept" label="continue">
    <go href="#third"/>
  </do>
  <p>
    <select title="search" iname="search_var" multiple="true">
      <option>name</option>
      <option>number</option>
    </select>
  </p>
</card>
<card id="third" title="Finish">
  <onevent type="ontimer">
    <go href="#first"/>
  </onevent>
  <timer value="20"/>
  <p align="center">
    You did it!
  </p>
</card>
</wml>

```

Appendix D

Tasks to perform during the userstudy.

- Task 1) Find a person named Smith by applying the search-function and give him a call!
- Task 2) You are going for a business trip over the weekend and you are staying at a hotel. Change your profile from Work to Travel!
- Task 3) Add a new number that can be used in your profiles!
- Task 4) Change your office-number to 123 123!
- Task 5) Delete your mobile number!
- Task 6) Add a new profile named "Work" and give it some optional numbers!
- Task 7) Edit your Home-profile so that the second number is diverted to your voicemail instead of your mobile!
- Task 8) Remove Travel from your list of profiles!
- Task 9) Change the Delay-time of the first number in your Hotel-profile!

Appendix E

Introductory userstudy 99-11-17

Which of the services of Personal Screen Call Server do you use?

- Scheduler
- Directory (Search)
- Message Diversion
- Call Log
- Personal Number
- Call Handler

How do you think the service works/ Do you miss any function?

Scheduler:

Directory:

Message Diversion:

Call Log:

Personal Number:

Call Handler:

Do you think you would use any of those services if they were available on a WAP-device?

Do you use any of the “Personal Number”-services from:

- GSM?
- DECT?

If “no”! Why not?

If “yes”! Do you think anything could be improved?

Appendix F

Final Result in Overview

Search (Directory): Input

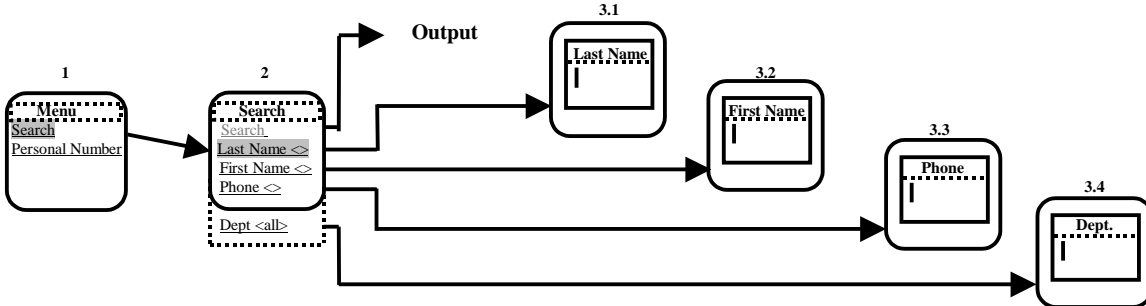


Figure 10.1. Search menu, final design.

Search (Directory): Search Result Output

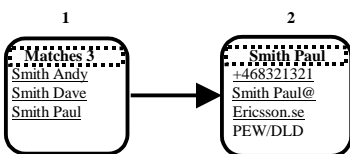


Figure 10.5. Output menu, final design.

Personal Number: Select Profile

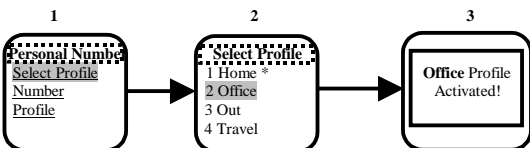


Figure 10.9. Select Profile menu, final design

Personal Number: Number/Profile navigation

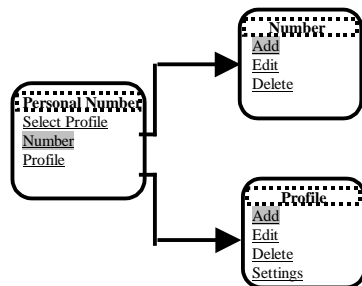


Figure 10.7. Separated Menus for the changing (add, edit, delete, [settings]) for Numbers and Profiles

Personal Number: Add Profile

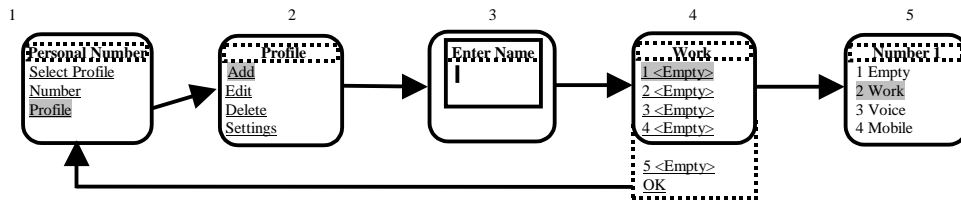


Figure 10.11. Add Profile with confirmation, final version

Personal Number: Edit Profile

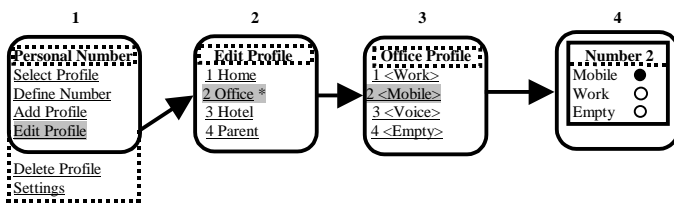


Figure 9.25. Edit Profile after redesign, also final version

Personal Number: Delete Profile

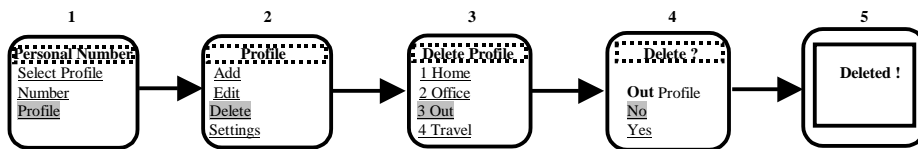


Figure 10.13. Delete Profile with confirmation, active Profile *not indicated*, final design

Personal Number: Profile - Settings - Delay

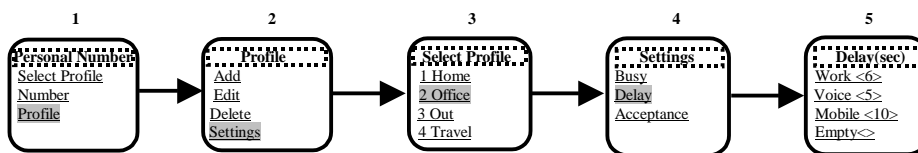


Figure 10.14. Settings with indication of unit used, final design