



University of Zurich
Department of Informatics



Diploma Thesis

May 15, 2008

Ontology-Based Cultural Personalization in Mobile Applications



Andreas Bossard
of Zug, Switzerland

Student-ID: 02-702-389
andreasbossard@gmail.com

Advisor: **Katharina Reinecke**

Prof. Abraham Bernstein, PhD
Department of Informatics
University of Zurich
<http://www.ifi.uzh.ch/ddis>

Acknowledgements

First of all I would like to thank Prof. Bernstein for giving me the opportunity to write my diploma thesis at his group and my advisor Katharina Reinecke who offered me to do this very interesting thesis and has supported me very well.

Moreover I would like to thank Mattias and Claudia for the screenshots of the foreign mobile phones and the helpful information and Vera and Keith for the proof-reading.

Last but not least with my studies coming to an end, I would like to say a big thank you to my parents who have always supported me and made my studies possible.

Abstract

This thesis analyzes how to automatically adapt mobile applications depending on the cultural background of the user. Concrete guidelines for the design of mobile applications for different cultures are suggested and the implementation of an adaptive prototype, which incorporates parts of those guidelines, is described. A domain ontology is used for defining the presentation of the menus and the navigation hierarchy structure and for assigning cultural dimensions to those components. A qualitative evaluation with Swiss and Chinese participants tries to validate some of the proposed guidelines and assesses the quality of the assignment of mobile phone menus to certain cultures. Furthermore it gives information about the usability of the user interface components of the prototype and indicates differences between the preferences of users from those two countries.

Zusammenfassung

Diese Diplomarbeit analysiert wie man mobile Applikationen automatisch an den kulturellen Hintergrund des Benutzers anpassen kann. Konkrete Richtlinien für die Gestaltung von mobilen Applikationen für verschiedene Kulturen werden vorgeschlagen und die Implementation eines adaptiven Prototypen, welcher Teile dieser Richtlinien berücksichtigt, wird beschrieben. Eine Domänen-Ontologie wird benutzt um die Darstellung der Menüs und der Struktur der Navigations-Hierarchie zu definieren und um diesen Komponenten kulturelle Dimensionen zuzuweisen. Eine qualitative Evaluation mit Schweizern und Chinesen versucht einige der vorgeschlagenen Richtlinien zu validieren und beurteilt die Qualität der Zuweisung der Mobiltelefon-Menüs zu gewissen Kulturen. Ausserdem gibt sie Aufschluss über die Bedienbarkeit der Komponenten der Benutzeroberfläche des Prototypen und deutet auf Unterschiede in den Präferenzen von Benutzern dieser beiden Länder hin.

Table of Contents

Table of Contents	ix
1 Introduction	1
1.1 Motivation	1
1.1.1 Culture influences cognition	1
1.1.2 Culture influences the performance of human computer interaction	1
1.1.3 Problem statement	1
1.1.4 Why a solution in form of a mobile application	2
1.2 Definitions	3
1.2.1 Culture	3
1.2.2 Cultural dimensions of Hofstede	3
1.2.3 User interfaces and culture	4
1.2.4 Localization	4
1.2.5 Difference between adaptable and adaptive	4
2 Related Work	5
2.1 Known user interface components for mobile applications	5
2.1.1 Differences between cell phone brands and countries	5
2.1.2 Types of known navigation menus and components	8
2.2 Adaptable/adaptive menus	10
2.3 Cultural preferences in user interface design	10
3 Guidelines for building a cultural adaptable cellular phone menu	13
3.1 Guidelines for building small screen menu interfaces	13
3.2 Chosen user interface components for testing the guidelines	16
4 Domain model ontology for culturally adaptive mobile applications	19
4.1 Why there is a need for a domain model ontology	19
4.2 Explanation of the domain model ontology	19
4.3 Options to assign the Hofstede values to the user interface components	20
5 CAMPA - a Culturally Adaptive Mobile Phone Menu Application	23
5.1 Requirements	23
5.2 Why Java Micro Edition?	23
5.2.1 The different implementation possibilities	23
5.2.2 Reasons for choosing Java Micro Edition	24
5.3 Evaluation of third-party libraries and frameworks	24

5.3.1	J2ME GUI libraries	24
5.3.2	XML parser	25
5.4	Architecture and program flow	25
5.4.1	Profile selection	25
5.4.2	The country questionnaire	26
5.4.3	Calculate the Hofstede values of the user	26
5.4.4	Show the menu and hierarchy that match the Hofstede values of the user	27
5.5	Assignment of Components and Hofstede values	28
5.5.1	The Menus and components	28
5.5.2	Assigning certain Hofstede scores to the components	31
5.6	Hardware requirements	32
6	User evaluation	35
6.1	Objectives	35
6.1.1	Chosen menus for the evaluation	35
6.1.2	Chosen navigation hierarchy for the evaluation	36
6.1.3	Hypotheses	36
6.2	Experimental methodology	36
6.2.1	Questionnaires	36
6.2.2	Evaluation sequence	37
6.2.3	User tasks	37
6.3	Pretest	37
6.3.1	Results	38
6.3.2	Conclusions	39
6.4	Main test	39
6.4.1	Results	40
6.4.2	Limitations	44
6.4.3	Conclusions	45
7	Limitations and Future Work	49
8	Conclusions	51
A	Appendix: Definition of income groups	53
A.1	Low income	53
A.2	Middle income	54
A.3	High income	55
B	Appendix: Navigation hierarchies of evaluated cell phones and CAMPA	57
B.1	Hierarchy structure of Sony Ericsson K800i	58
B.2	Hierarchy structure of Nokia 6131	59
B.3	Hierarchy structure of CAMPA for evaluation	60
C	Appendix: Screenshots of cell phones	61
C.1	All screenshots of Sony Ericsson K800i	61
C.2	All screenshots of Nokia 6131	63
C.3	Screenshots of DoCoMo FOMA N905i	65

D	Appendix: Forms for the user evaluation	67
D.1	Notes sheet	68
D.2	Task sheet	69
D.3	User questionnaire	81
E	Appendix: Screenshots of the menus used for the evaluation	85
F	Appendix: Installation notes	87
G	Appendix: Enclosed CD	89
	List of Figures	91
	List of Tables	93
	Bibliography	95

1

Introduction

1.1 Motivation

1.1.1 Culture influences cognition

Several studies conducted by Nisbett have shown that the way we perceive the world is influenced by our cultural background. East Asians were more sensitive to contextual changes than to focal object changes. For example, when shown changes in pictures, East Asians noted more changes in the background and Westerners noted more changes at the focal object in the foreground [Nisbett, 2003; Masuda and Nisbett, 2006]. In addition Westerners tend to group objects in taxonomic ways whereas East Asians tend to group objects depending on their relation to each other. For instance when asked to group a cow with a chicken or a cow with grass, Westerners tended to group the cow with the chicken, because they are both animals (taxonomic view), whereas East Asians tended to group the cow with the grass, because the cow eats grass (relational view) [Nisbett, 2003].

1.1.2 Culture influences the performance of human computer interaction

As shown by research on culturally adapted web sites and interfaces, culture has an impact on certain preferences in web design [Barber and Badre, 1998; Burgmann et al., 2006] and on performance of users [Badre, 2000; Reinecke, 2005]. Heimgärtner found that human computer interaction between different cultures differs significantly [Heimgärtner, 2007a; Heimgärtner, 2007b]. The survey of Van Biljon showed that among other cultural dimensions uncertainty avoidance and individualism influence mobile phone usage [Van Biljon, 2006].

Therefore the goal must be to adapt the user interface to the cultural background of the user. Doing this manually is very time consuming [Reinecke, 2005]. So one proposed solution is the use of interfaces that automatically adapt to the cultural background of the user [Reinecke and Bernstein, 2007]. The first adaptation should be handled automatically by the application. Further changes of the user interface should underlie computer supported adaptations, which means that the user can intervene if proposed changes are not desired. Reinecke mentions the intention to develop a prototype system to test the most promising adaptivity algorithms. The Culturally Adaptive Mobile Phone Menu (CAMPa) which is introduced in this thesis is such a prototype.

1.1.3 Problem statement

This thesis tries to tackle the problem of adapting a mobile phone application to the cultural background of the user. As chapter 2 will show there is no such adaptive application yet. A particular challenge is to translate the results from

research of cultural differences in human computer interaction on desktop computers to possible differences in mobile applications.

1.1.4 Why a solution in form of a mobile application

There are two reasons why a solution in form of a mobile application was developed. Firstly, a lot of people in developing countries do not have access to a computer, but they do have access to a mobile phone. This is shown in the charts in figure 1.1. The number of mobile phone subscribers per 1000 people is clearly growing faster than the number of personal computers per 1000 people for both people with middle income and people with low income. In 2004 there were 294 mobile phone subscriptions per 1000 people in middle income countries, but only 61 personal computers. Low income countries had 42 mobile phone subscriptions per 1000 people compared to 11 personal computers. The boom of mobile phone subscriptions in middle income and low income countries is even more evident when looking at figure 1.2. By 2004 there were 3.73 mobile phone subscriptions per personal computer in low income countries and 4.82 mobile phone subscriptions per personal computer in middle income countries and these ratios are rising sharply since about the year 2000. Definition of the income groups can be found in appendix A.

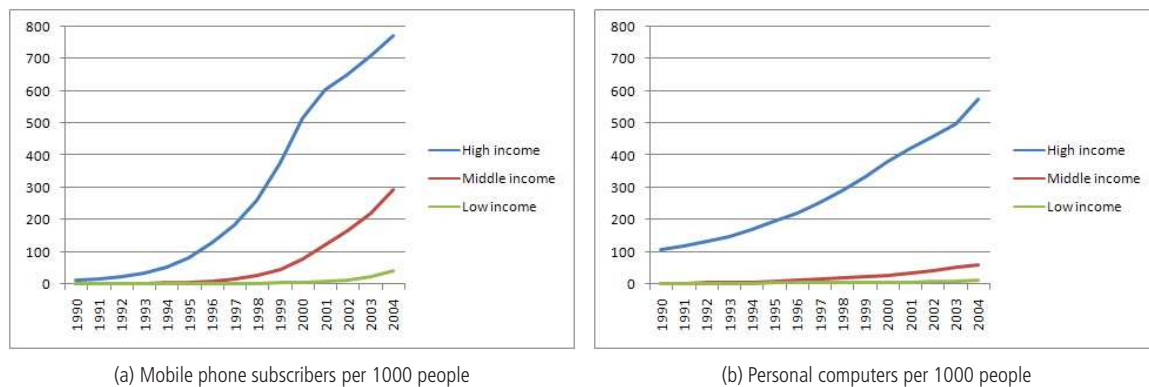


Figure 1.1: Diagrams showing the change in mobile phone subscribers and personal computers (from [ITU, 2006])

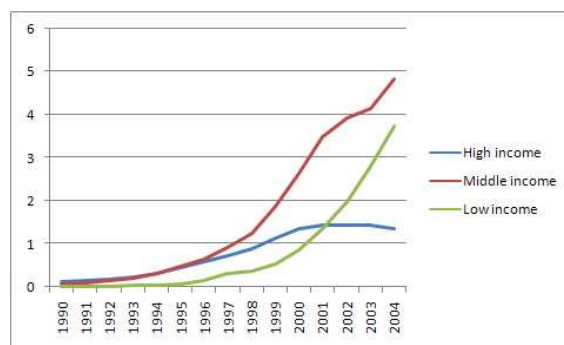


Figure 1.2: Chart showing the number of mobile phone subscribers per personal computer (from [ITU, 2006])

In order to evaluate whether it is possible to give the user a culturally adapted interface, it makes more sense to present that interface on a device he is already familiar with - such as a cell phone - instead of a desktop computer.

Secondly, this solution helps to show that the user model can be reused. Another diploma thesis implemented an adaptive website to display in a desktop browser. In a later version of this mobile application, both the desktop application and the mobile application could access the same user model. This could show that the user model can be stored in a central place and used by different applications.

1.2 Definitions

1.2.1 Culture

There are many definitions of culture, whereas one definition by Marsella and Yamada [Marsella and Yamada, 2000] is particularly interesting:

Shared learned meanings and behaviors that are transmitted from within a social activity context for purposes of promoting individual/societal adjustment, growth, and development. Culture has both external (i.e. artifacts, roles, activity contexts, institutions) and internal (i.e., values, beliefs, attitudes, activity contexts, patterns of consciousness, personality styles, epistemology) representations. The shared meanings and behaviors are subject to continuous change and modification in response to changing internal and external circumstances.

Further Marsella and Yamada state that culture is the lens or template we use in constructing, defining and interpreting reality. This definition suggests that people from different cultural contexts will define and experience reality in very different ways [Marsella and Yamada, 2000]. This is consistent with the studies, which are mentioned in section 1.1.1.

1.2.2 Cultural dimensions of Hofstede

Geert Hofstede analysed data of IBM workers from different countries and developed a model that differentiates five culture dimensions [Hofstede, 2001]. These five dimensions are also used in the Cultural User Model Ontology [Reinecke et al., 2007] and in the domain model ontology which both are used in the prototype of this thesis. Therefore the meaning of the five dimensions will be explained here shortly [Marcus, 2002]. Note that even though Hofstede's model is well established, there are also criticisms of the model [Marcus, 2006].

Power distance (PDI): The extent that people accept large or small distances of power in social hierarchies.

Individualism vs. collectivism (IDV): The orientation to individual or group achievements.

Masculinity vs. femininity (MAS): The degree to which a culture separates / does not separate traditional gender roles.

Uncertainty avoidance (UAV): The degree to which a culture is uncomfortable with uncertainty.

Long-term time orientation (LTO): The orientation to Confucian thought, which emphasizes patience.

1.2.3 User interfaces and culture

There are two categories of topics that are affected in human computer interaction localization [Heimgärtner, 2007b]:

1. Presentation of information (e.g. color, time and date format, icons, font size) and language (e.g. font, writing direction)
2. Dialog design (e.g. menu structure and complexity, layout, widget positions) and interaction design (e.g. navigation concept, system structure, interaction path, interaction speed)

This thesis covers the second category. Reasons for not including the first category can be found in the Limitations in Chapter 7.

1.2.4 Localization

Localization is the process of modifying products or services to account for differences in distinct markets [LISA, 2007].

1.2.5 Difference between adaptable and adaptive

In an adaptable application the user controls the adaptation. Whereas in an adaptive application the adaptation is controlled by the system [Findlater and McGrenere, 2004].

2

Related Work

2.1 Known user interface components for mobile applications

2.1.1 Differences between cell phone brands and countries

Different current cell phone models which are available in Switzerland were evaluated to determine the current state of the art of cell phone menus. Some foreign and older cell phones were also explored so far as information about them was available.

Evaluated cell phones

The following cell phones currently sold in Switzerland were evaluated in particular. They have the same screen resolution. This is crucial, because a higher screen resolution increases the display possibilities. Section 2.1.2 references to these cell phones. The available menus and components are explained in more detail in that section.

Sony Ericsson: The device evaluated was a Sony Ericsson K800i with Vodafone firmware by swiss mobile telecommunication operator Swisscom. It uses grids for the main menu (see figure 2.1a); and lists (see figure 2.1b) and tabs (see figure 2.1c) for the submenus. The K800i has a screen width of 240 pixels and a screen height of 320 pixels [Sony Ericsson, nd]. It uses a lot of icons. When focused, the icons for the menu items in the main menu are animated and the name of the function is shown in the title bar. The order of the menu items in the main menu can not be changed. More screenshots are provided in appendix C.1.

Nokia: The evaluated Nokia 6131 also had a Vodafone firmware, this explains why the icons in the screenshots look similar to those of the Sony Ericsson K800i (see appendix C.2). The navigation hierarchy structure was compared to a Nokia 6131 without Vodafone firmware and there were only slight differences, so the Vodafone specific menu items were not taken into account in the navigation structure shown in the appendix. The Nokia 6131 uses a grid menu as default main menu layout with animated icons and list menus for submenus (see figures C.2a, C.2f, C.2h and C.2i). The style of the main menu can be changed into list or tab menu (see figures C.2c and C.2d). There are two grid styles available. One shows icons and text, the second one shows only icons and the menu item name in the title bar (see figures C.2a and C.2b). The order of the items in the main menu can be freely defined by the user. The Nokia 6131 has a screen resolution of 240 x 320 pixels [Nokia, nd] and also uses a lot of icons as with the Sony Ericsson K800i.

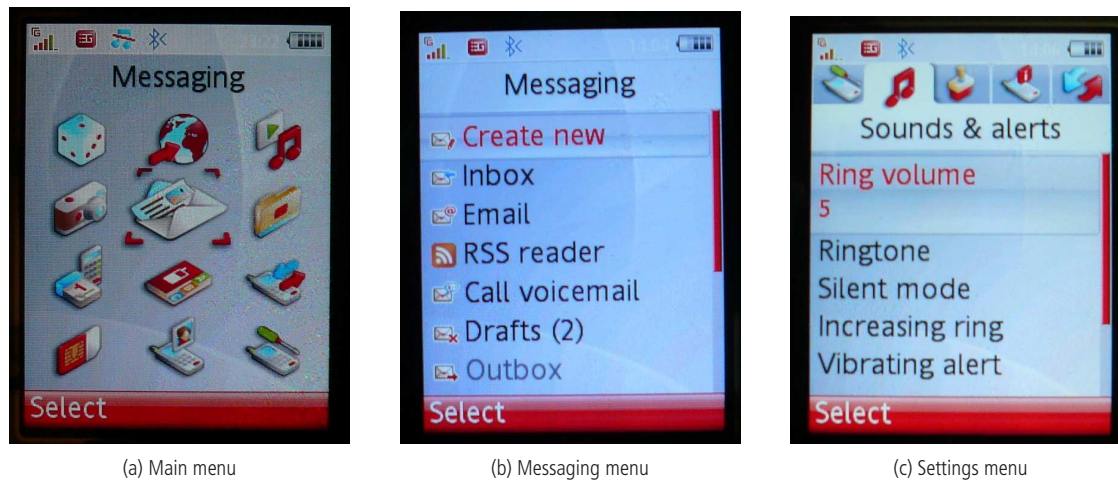


Figure 2.1: Screenshots of Sony Ericsson K800i

Foreign cell phones

Evaluation of foreign cell phones was much harder, because naturally there are not many people in Switzerland who have such cell phones. The following cell phones are not sold in Switzerland, however, the following section should give an impression of some foreign cell phones.

au by KDDI: KDDI is a Japanese telecommunication operator. It was possible to get access to a Sanyo A5514SA for a short evaluation, but unfortunately, not for a comprehensive evaluation. Some screenshots of the Sanyo A5514SA give an impression of the user interface. There are numbers next to the menu items which indicate direct access via keypad (see figure 2.2a). There is not much use of icons and if there are icons, they are not colorful. The main menu contains animated icons (see figures 2.2b, 2.2c), but they are all in blue. The data folder menu also contains has icons next to each folder. In the data folder list there are cues, which indicate how many elements a folder contains (see figure 2.2d). The phonebook gives the possibility to display a photo of each contact which is not the case for the Swiss cell phones (see figure 2.2e). The tabs in the phonebook group people whose names start with the same syllable. This is useful in Japan, because a lot of names start with the same syllable (C. Bretscher, personal communication, March 31, 2008).

DoCoMo FOMA: The DoCoMo FOMA phones are Japanese cell phones. Unfortunately there was no DoCoMo FOMA cell phone available for evaluation. However, some examples of the user interfaces can be found in the user manuals for the cell phones. For example in the manual of N905i [NTT DoCoMo, 2007]. The screenshots taken from the manual can be found in appendix C.3. Noticeable are the numbers in the menus to indicate fast access to the functions and the simple design (see figure C.3a). There are only a few icons used it seems. The only icons from all screenshots in the manual were found in the contact detail screen of the phonebook (see figure C.3b). In the phonebook of the N905i there are tabs to filter the phonebook-entries (see figure C.3c). One tab type has Japanese syllables, another has numbers for filtering by memory number and one has numbers for filtering by group number.

Anycall: Anycall is a South Korean cell phone brand and belongs to the Samsung group. Upon request Samsung kindly provided some screenshots of the Anycall SCH-V840 (see figure 2.3). It has the same screen resolution as the Sony Ericsson and Nokia cell phones evaluated [Shenzhen Keenmas Trading Co., nd]. The displayed menu is three-dimensional and has three levels. Each level contains a ring of three functions which is rotatable. The background



Figure 2.2: Screenshots of Sanyo A5514SA

color changes when another level is highlighted. When the top level is highlighted the background is red, the middle level has a green background and the bottom level a blue background. The menu is much more colorful than the Swiss or Japanese menus. According to Mr. Piccand from Samsung the icons are much more animated than the menu icons in Swiss cell phones (Y. Piccand, Samsung, personal communication, May 2, 2008).

Statements of mobile phone manufacturers

Besides Samsung several other companies were queried about whether the navigation menus of their mobile phones vary between different countries. There was no feedback from LG Electronics and Motorola. Nokia (P. Preuss, Nokia, personal communication, March 11, 2008) was not able to provide any information. Sony Ericsson stated, that their menu navigation does not differ between different markets, but they "do specific menu structures for operators but it is still the same paradigm" (E. Nilsson, Sony Ericsson, personal communication, March 28, 2008). Screenshots were not provided by any of the companies who responded.

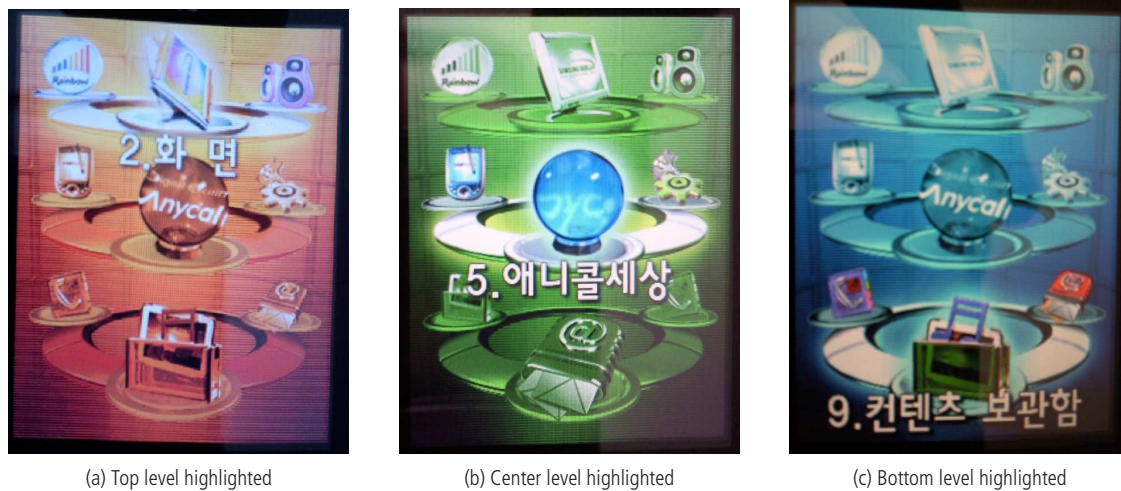


Figure 2.3: Screenshots of Anycall SCH-V840

2.1.2 Types of known navigation menus and components

In this section several menu and component types that were found on the evaluated and foreign mobile phones in section 2.1.1 or in different mobile applications are introduced to show what the current state of the art in the field of navigation menu design is.

List menus

List menus have the advantage that they do not need icons and therefore no high resolution display is needed to use them. Moreover they can already be navigated, using a device that only supports up, down and enter input. Unlike tab menus and grid menus, which need icons and also a device that supports left and right input. All evaluated cell phones use list menus. Nokia uses them on all levels and Sony Ericsson not for the entry-menu.

Grid menus

Many of the evaluated cell phones use grid menus for the entry-menu only, but not for navigation in submenus. This is probably the case because grid menu items always need icons. It can be hard to design icons for functions on lower levels, because not every function can be expressed by a meaningful icon. Briefly said: "At lower levels the selection choices are mostly too specific to be unambiguously represented" [Studio 7.5, 2006]. How many rows and columns a grid menu has depends on the resolution of the mobile phone. The evaluated cell phones had 3 columns and 3 or 4 rows. On the Sony Ericsson K800i there are constantly 12 items in the main menu. On the Nokia 6131 the number of items in the main menu is variable and can be defined by the user. If there are too many items to fit on one screen, a scroll-bar is shown.

Possible enhancements: More rows and columns would make it possible to place more functions on the screen, but this leads also to smaller icons, which are maybe harder to associate with the corresponding functions.

Tab menus

Tab menus are useful to display a few categories with a lot of list-items in them. The user can quickly go through each tab (each category) and see which items it contains. Therefore it is quite easy to explore the menu hierarchy. Tab menus are a possibility to display two navigation levels (categories and their entries) on one screen. Because of that the user has a cue where he is located inside the navigation structure.

The downside is as with grid menus in that there is not much space in the tabs, thus only icons can be used. (Or in Asian countries word-signs could be used). All of the evaluated cell phones use tab menus.

Search menus

This idea is similar to the desktop search on desktop computers. You do not move through the menu hierarchy, instead you search for the desired function by entering text into a searchbox. This idea was proposed by [Jones and Marsden, 2006]. Their search is based on a B+Tree. For making searching easier, they added following feature: When pressing two times button "1" that could mean "AA" or "CA". The program automatically searches for all the different possibilities.

Search menus are not available on any of the evaluated cell phones. The most similar component to a search menu found is a fast search functionality for the address book on the Sony Ericsson K800i (see figure C.1f). As the user types the first characters of a name, the address book jumps to the contact entry whose name starts with the entered characters. Another similar component is the contacts filter in the addressbook of the Nokia 6131 (see figure C.2e). As the user types the addressbook is filtered. Only those contacts whose names start with the same characters as the entered text are displayed.

Possible enhancements: Functions could be tagged with different keywords, so the users do not need to know the exact name of the function.

Other mobile search-applications: K-menu [Lee and Lee, 2007] and Qix [Zi Corporation, 2008] have such a search-navigation functionality.

Tree menus

Tree menus are very rarely used on cell phones. However, there are some applications for smartphones and PDAs. There are several file browsers like Resco Explorer for Pocket PC, Smartphone or Palm OS [Resco, nd]; FileZ File Browser for Palm OS by nosleepsoftware [nosleepsoftware, nd] or X-plore by Lonely Cat Games for Symbian [Lonely Cat Games, nd]. Another application called "Handy Tree Notes" by PPC Utils [PPC Utils, nd] can be used for Windows Mobile for creating trees of notes and to-do lists. Tree menus are not used on any of the evaluated cell phones.

Components of user interfaces

Numbers in lists: Japanese cell phones seem to make a lot of use of numbers next to the menu items to indicate direct access via keypad. None of the evaluated cell phones show numbers in lists.

Icons in lists: All evaluated cell phones make heavy use of icons in their interfaces. Japanese cell phones use icons only rarely.

Breadcrumb trail: Breadcrumb trails give the user more information about his current position in the hierarchy. None of the cell phones evaluated uses breadcrumb trails. Some older Nokia phones had number breadcrumb trails which were used for fast function access, as with the Nokia 3330.

Help text: Help text provides the user with more information about a function before he selects it. All evaluated cell phones had the possibility to display help texts. Normally this is done in a pop-up window. The Sony Ericsson K800i displays help text at the request of the user (see figure C.1g). The user has to press the help soft key. The Nokia 6131 shows the help text automatically after a menu item was highlighted for about 15 seconds (see figure C.2g).

Cues, which flag category-functions: Category cues are icons, which are displayed next to a menu item to indicate, that this menu item is a category and a selection of it does not start up a function, but shows the sub menu items. None of the cell phones evaluated showed category cues. The Sanyo A5514SA shows something similar (see figure 2.2d). The file folders have a cue that shows the number of files in each folders.

Navigation hierarchies: Parts of the menu structure hierarchies of the evaluated devices were explored. Refer to the figures in Appendix B to see a visualization of the menu structures. Both hierarchies have a hierarchy level maximum of about four levels. Striking is that in the Nokia 6131 there is mostly only one way to reach a function, whereas in the Sony Ericsson K800i there are a lot of functions which can be reached in two ways (for example "Message Alert").

2.2 Adaptable/adaptive menus

Adaptive and adaptable menus were evaluated for desktop applications by Findlater and McGrenere [Findlater and McGrenere, 2004] and Park et al. [Park et al., 2007]. Findlater and McGrenere showed that static menus are faster than adaptive menus and adaptable menus are as fast as static menus, except when the adaptable menu was used first, then adaptable menus were as slow as adaptive menus. But the static menu used in this experiment relies on the unrealistic assumptions that the system knows the selection frequency of items beforehand and that the selection frequency stays constant. The adaptable menu was preferred to the static menu, but not to the adaptive menu. But the participants that preferred the adaptive menu expressed very strong support for it. Therefore the authors recommend to use a combination of adaptable and adaptive menus. For example by proposing the user frequently used items for adaptation.

Park et al. [Park et al., 2007] found the adaptable menu to be faster and preferred compared to adaptive and traditional static menus. The downside of adaptable menus are the efforts of the users to reorder the lists. Therefore they recommend also a combination of adaptable and adaptive menus, for example by highlighting frequently used items as indication for the user to reorder the list.

Research about adaptation for mobile applications was for example done by Goren-Bar and Kuflik [Goren-Bar and Kuflik, 2004]. They showed, that users assigned significantly higher System Usability Scale scores to an adaptive system than to a non-adaptive system. Even though they ran their experiment not on an actual cell phone, but in a simulation in MS Mobile Explorer.

2.3 Cultural preferences in user interface design

Research on preferences in user interface design between different countries and combining these with Hofstede values concentrated mainly on desktop applications [Marcus et al., 1999; Gould et al., 2000; Marcus, 2001; Marcus and Gould, 2001; Marcus, 2002; Marcus et al., 2003; Ford and Gelderblom, 2003; Marcus, 2005; Röse, 2005a; Burgmann et al., 2006; Kamenz, 2006].

An exception is the research of Cha et al. and Choi et al. [Cha et al., 2005; Choi et al., 2005]. They explored preferences in mobile user interfaces of participants from different countries. Cha et al. created mobile user interfaces in a flash application. The participants used it on a desktop computer. Choi et al. filmed the use of certain mobile phone services and showed these videos to the participants. The participants then answered interview questions about certain properties of the shown user interfaces. The results of these studies are summarized in section 3.1.

A prototype similar to the one presented in this thesis was not found. Related research either concentrated on cultural differences between desktop applications or the mobile application was emulated in a desktop application.

3

Guidelines for building a cultural adaptable cellular phone menu

3.1 Guidelines for building small screen menu interfaces

A lot of research was done to analyze differences between user interfaces in different countries and combine these differences with Hofstede values of the countries [Marcus et al., 1999; Gould et al., 2000; Marcus, 2001; Marcus and Gould, 2001; Marcus, 2002; Marcus et al., 2003; Ford and Gelderblom, 2003; Marcus, 2005; Röse, 2005a; Cha et al., 2005; Choi et al., 2005; Burgmann et al., 2006; Kamentz, 2006]. A part of this research was combined into general guidelines for designing user interfaces [Reinecke, nd]. The following table 3.1 shows concrete guidelines for mobile devices. Where not otherwise referenced, ideas for the concrete guidelines were derived from [Reinecke, nd]. These concrete guidelines take into account the restricted display possibilities of mobile devices. For example the small screen prevents displaying of a lot of information on one screen.

General Guidelines	Hofstede Classification		Concrete Guidelines
Show breadcrumb trail [Marcus, 2005]	LTO	Low	Show breadcrumb trail. Generally, show position in the hierarchy.
Much information at first level, only few levels	LTO PDI	High High	<ul style="list-style-type: none"> • Show a lot of menu items at the first level of the navigation. • Use a shallow navigation hierarchy. This could be achieved for example by using ListMenu in combination with a shallow hierarchy or by using SearchMenu, which represents no hierarchy.
Show only few information at the first level, hierarchy can be deep	LTO PDI	Low Low	<ul style="list-style-type: none"> • Show only a few menu items at the first level of the navigation. • Use a deep navigation hierarchy.
Let the system initiate communication	PDI	High	If activated, helptext is shown automatically.
User can initiate communication	PDI	Low	If activated, helptext is shown after the user pressed a button.
Low image-to-text ratio	IDV	High	Use menus that can be shown without icons. For example ListMenu, SearchMenu or TreeMenu.
High image-to-text ratio	IDV	Low	Use menus that have icons as an inherent property. For example GridMenu, or TabMenu. Add icons to lists of ListMenu and TabMenu. If a breadcrumb trail is used, show the one with icons.
Contrast between interface areas	IDV	High	Use background color for contrasts. Possible interface areas could be the title bar respectively the breadcrumb trail, the area containing the menu items and the menu bar which contains the command labels. Menu items belonging together could be grouped by using the same background color.
Apply a high multimodality to the interface	IDV	Low	Use additional output-possibilities like sounds and vibration to give the user cues about how deep inside the navigation he is.
Complex interface possible	UAV	Low	Use menus with a lot of menu items and add a lot of user interface components like icons and breadcrumb trail.
Simple user interface	UAV	High	Use menus with only few menu items, as few user interface components as possible.

Show the position of the user in the hierarchy	UAV	High	Use menus with breadcrumb trails. Maybe use a tree menu.
Use redundant cues	UAV	High	E.g. use a different background color and bold font to signal the importance of a navigation item
Use non-redundant cues to maximize information	UAV	Low	E.g. Use only bold font for the meaning "important". The background color can signal other meanings.
Show large amount of information within a screen, use the screen space efficiently [Cha et al., 2005; Choi et al., 2005]	UAV	High	Use as many user interface components as possible. For example use icons for navigation items to make their meaning clearer and use breadcrumb trails.
Show secondary information [Choi et al., 2005]	UAV	High	For example use helptexts or icons to give more information about choices.
Show only limited choices [Marcus, 2002; Reinecke, nd]	MAS UAV PDI	High High High	Try to limit the number of choices as much as possible. E.g. a GridMenu with max. 6 navigation items.

Table 3.1: Concrete guidelines for designing multi-cultural mobile user interfaces

There is a potential conflict between "Simple user interface" and "Show only limited choices" for users with high uncertainty avoidance score and "Show large amount of information within a screen" for users with high uncertainty scores.

Marcus and Gould [Marcus and Gould, 2001] recommended to use simple interfaces with limited choices for users with low uncertainty avoidance score, after comparing websites from different countries. They described Simplicity as follows: "with clear metaphors, limited choices, and restricted amounts of data" and Complexity as: "with maximal content and choices"

This recommendation contravenes the results of the research of Cha et al. and Choi et al. [Cha et al., 2005; Choi et al., 2005]. They did research on the cultural difference in mobile user interfaces. Cha et al. found out that Korean participants (high uncertainty avoidance score) preferred the cluttered interfaces (complex) and participants from the United Kingdom (low uncertainty avoidance score) preferred interfaces with limited choices (simple) [Cha et al., 2005].

Choi et al. detected that participants with high uncertainty avoidance score preferred a large amount of information within a screen and efficient layout or space usage, so they do not need to scroll and have more knowledge about what the impact of the next selection will be [Choi et al., 2005].

In consideration of the fact, that Cha et al. and Choi et al. concentrated on mobile applications, their recommendation to show a lot of information on one screen for users with high uncertainty avoidance seems to be more suitable and will be used in the domain model.

3.2 Chosen user interface components for testing the guidelines

Based on the concrete guidelines of section 3.1, different menu components were taken into consideration to research the mapping of Hofstede scores and components. Following these components are listed with an explanation why they were chosen. For a more elaborate and technical explanation of the user interface components and their functionality refer to section 5.5.1. An overview of these assignments of components and the Hofstede values is given in figure 3.2.

ListMenu: Lists are very simple user interfaces and use the screen space relatively inefficiently (in comparison with the TabMenu for example). Therefore they should be suitable for users with low uncertainty avoidance score [Cha et al., 2005; Choi et al., 2005].

GridMenu: The GridMenu has a high image to text ratio, which should suit users with a low individualism score [Reinecke, nd]. The GridMenu uses screen space relatively inefficiently because the big icons need a lot of space, which indicates suitability for users with low uncertainty avoidance score. On the other hand the icons provide additional information, therefore GridMenu should be suitable for people with somewhat low uncertainty avoidance [Cha et al., 2005; Choi et al., 2005].

TabMenu: The TabMenu is more complex than the other menus. Pressing the navigation key left and right reveals information formerly hidden in the other tabs. There is also more information about the hierarchy (two levels of the hierarchy can be seen at the same time). These attributes should make the TabMenu appropriate for users with high uncertainty avoidance [Cha et al., 2005; Choi et al., 2005].

SearchMenu: The SearchMenu inherently has no hierarchy, because potentially all the functions can be displayed in the search selection pop-up. And a selection leads directly to the function, so no navigation through a hierarchy is required. Accordingly the SearchMenu should be suitable for users with high long term orientation scores and high power distance scores [Reinecke, nd]. This menu is very simply arranged, because it contains only a search text field and the space usage is not especially efficient, compared for example with the TabMenu. Hence the SearchMenu

Menus	LTO	PDI	IDV	UAV	MAS
ListMenu				low	
GridMenu			low	low	
TabMenu				high	
SearchMenu	high	high		low	
TreeMenu	low				
User interface components					
No breadcrumb	high		low	low	
Breadcrumb (String or Number)	low		high	high	
Breadcrumb (Icons)	low		low	high	
No helptext				low	
Helptext automatic		high		high	
Helptext by request		low		high	
No icons in lists			high	low	
Icons in lists			low	high	
Numbers in lists				high	
No cues, which flag categories				low	
Cues, which flag categories				high	
Hierarchy					
Deep Hierarchy	low	low			
Shallow Hierarchy	high	high			
Show only a few navigation-items on screen		high		high	high
Show a lot of navigation-items on screen		low		low	low

Table 3.2: Rough overview of assignment of components and Hofstede values

should be suitable for users with low uncertainty avoidance. But for efficient search, every menu item has keywords which are shown. Those keywords provide additional information, hence the SearchMenu possibly is must suitable for users with somewhat low uncertainty avoidance [Cha et al., 2005; Choi et al., 2005].

TreeMenu: The TreeMenu uses the screen space not very efficiently and this would qualify it for users with low uncertainty avoidance score [Cha et al., 2005; Choi et al., 2005], but the user has information about his position inside the hierarchy, which is suitable for users with high uncertainty avoidance [Reinecke, nd]. This should even out the effects of screen space usage.

When the user has a cue about the position in the hierarchy, this should suit short term orientated users. [Marcus, 2005]

Numbers in lists: Numbers which imply the functionality of direct choice via keypad, add more information to the screen and therefore they should be suitable for users with high uncertainty avoidance score [Cha et al., 2005; Choi et al., 2005].

Icons in lists: The more icons are shown in a menu, the higher the image to text ratio. Users with low individualism score prefer a high image to text ratio in user interfaces [Reinecke, nd]. By adding icons, more information is added

to the screen, which should make it more suitable for users with a high uncertainty avoidance score [Cha et al., 2005; Choi et al., 2005].

Breadcrumb trail: The breadcrumb trail component shows users their current position and adds additional information to the user interface what should please users with high uncertainty avoidance score [Reinecke, nd; Cha et al., 2005; Choi et al., 2005]. Moreover, users with low power distance index should prefer user interfaces with breadcrumb trails [Reinecke, nd]. Furthermore breadcrumb trails consisting of icons increase the image to text ratio which should suite users with low individualism [Reinecke, nd].

Help text: Help text allows the user to get more information about a function, before he selects it, which should be preferred by people with high uncertainty avoidance [Choi et al., 2005].

The help text can be shown automatically or by request of the user. If the system initiates communication by showing help texts automatically this should please to users with high power distance index. Users with low power distance index would like to initiate communication themselves [Reinecke, nd].

Cues, which flag category-functions: When this component is used in the interface some functions will have an icon that marks them as category-functions, which means, that they do not execute a function, but just are categories which have more functions beneath them in the hierarchy. This should help users with high uncertainty avoidance to get more information about the consequences of selecting the function [Choi et al., 2005].

Navigation hierarchies: Different hierarchies can be defined. So it is possible to use a shallow or a deep navigation hierarchy, depending on the long term orientation and power distance index of the user [Reinecke, nd]. It is also possible to define how many menu items are shown on each level of the hierarchy. The more menu items are present on the screen, the more choices and the more information is available, which should suit users with low masculinity score [Marcus, 2002] and high uncertainty avoidance score [Cha et al., 2005; Choi et al., 2005].

Note that other factors can influence how well someone navigates in a hierarchy. Age for example can have an influence. In an experiment conducted by Ziefle and Bay [Ziefle and Bay, 2004] a fourth of the older participants (between 50 and 64 years old) did not have a hierarchical mental representation of the menu, whereas all of the younger participants (between 20 and 32 years old) had a hierarchical mental representation of the menu structure. Participants without a correct mental model had a significant lower navigation performance.

Another factor, which can influence the mental representation of the menu is literacy as suggested by Walton et al. [Walton et al., 2002; Walton and Vukovic, 2003]. The concept of information hierarchies was not clear for all participants. When asked to draw a family tree, one participant did not draw an hierarchical family tree but seemed to draw two lineages, one maternal and one paternal. Only a tenth of the participants were able to interpret breadcrumbs.

4

Domain model ontology for culturally adaptive mobile applications

Besides the Cultural User Model Ontology CUMO, which links the user with the Hofstede values, there is the domain model ontology, which links the user interface components with the Hofstede values. The first part of the chapter gives an explanation, why a domain model ontology is needed, the second part explains the domain model ontology in detail.

4.1 Why there is a need for a domain model ontology

An ontology is needed to link the modules of the user interface to the cultural values of Hofstede. For example the ontology allows the application to make a connection between the TabMenu and certain Hofstede values.

4.2 Explanation of the domain model ontology

The domain model ontology is visualized in figure 4.1.

The class PersonPreference: Instances of PersonPreference hold the preferences of the users. The name, the Hofstede scores, the favorite menu and the favorite hierarchy of the user after the first adaptation are saved by CAMPa as PersonPreference individuals. The Hofstede values were directly integrated as properties into the class, instead of linking to the HofstedeValues class. This was done with the requirement in mind, that further adaptations of PersonPreference should be possible by the application. J2ME has no external library for easy editing of OWL and therefore with the current solution, changing of the values of PersonPreference individuals is kept as simple as possible.

The class Menu and its subclasses: The class menu can not have individuals, so it is merely a template class. Its properties are inherited by every subclass. It contains five subclasses, which represent the five menu types. A menu is defined by creating an individual of a subclass. Such a menu can have several components, which can be activated or deactivated by defining the corresponding properties in the individual. For example the icons can be activated by setting the property hasIconsInList to true. Menus are linked with a HofstedeValues individual.

The class Hierarchy: An individual of the class Hierarchy links to the top FunctionItems of a navigation hierarchy by the property hasTopMenuItems. The lower levels of the hierarchy tree are defined by the children items in those FunctionItems. A hierarchy is also linked with a HofstedeValues individual.

The class FunctionItem: The class FunctionItem contains all information of an item in the menu. To define a menu item, an individual of the class FunctionItem is created. The property hasIconFileName points to the file name of an icon for this menu item. The property hasKeywords can contain some keywords, which are only used for SearchMenu. If there were no keywords, the user would have to know parts of the function name to find the function. The property hasHelpText, contains a help text which is only used, if the help text functionality is activated in the menu, which contains this menu item. Finally the property hasChildrenItems links to a list of FunctionItems. If the list is empty, this FunctionItem is a leaf FunctionItem, that is it links to a function. On the other hand if the FunctionItem has children items it is a category FunctionItem.

The class HofstedeValues: An individual of the class HofstedeValues contains the cultural dimensions of Hofstede in five properties: hasIndividualism, hasLongTermOrientation, hasMasculinity, hasPowerDistance and hasUncertaintyAvoidance. If one of those five scores is not set, the average score is taken. By linking individuals of other classes with an individual of this class, it is possible to assign Hofstede scores them.

The classes HelpTextType and BreadcrumbType: Individuals of these classes serve as constants for properties of menus.

4.3 Options to assign the Hofstede values to the user interface components

There were two options how to assign the Hofstede values to the user interface components. Advantages and disadvantages of the two options will be shortly evaluated to justify the chosen option.

Option 1: For each menu type there are multiple individuals in the domain model. For example there is a ListMenu with icons and breadcrumb trail or a ListMenu with icons, but without breadcrumb trail. These would be two different individuals in the ontology .

Option 2: For each menu type there is one individual, which has certain Hofstede values associated with it. When adding user interface components like icons or breadcrumb trail these Hofstede values are decreased or increased. For example adding icons would decrease the individualism score of the ListMenu.

Option 1 was chosen because of the following reasons: The most important advantage of option 1 is that the domain ontology is still easy to adapt. One of the requirements of the application is that the Hofstede values in the domain ontology should always be changeable. With option 2 it is hard for the application to change the Hofstede values in the domain ontology. This argument can be explained with a scenario of adapting the user interface: For example if several users have a high uncertainty avoidance and have no problems to navigate in a menu, which is designed for low uncertainty avoidance, the uncertainty avoidance value of that menu needs to be increased. If that menu has an certain instance like in option 1 it is no problem to change its Hofstede values, but in option 2 the application could only change the Hofstede values of certain components, but not of the whole menu.

Furthermore with this approach, it is easier to get an overview of which menus have which optimal Hofstede values before the application is executed. Whereas with the second option, Hofstede values of the final menu are not calculated before runtime.

The main disadvantage of this approach is, that when a new component is introduced into the domain model, the Hofstede values of the already defined menu instances maybe have to be changed and recalculated. Whereas with option 2 for each additional menu component only has to be defined how much it changes the Hofstede values of the menu, and the rest of the calculation would be done by the application.

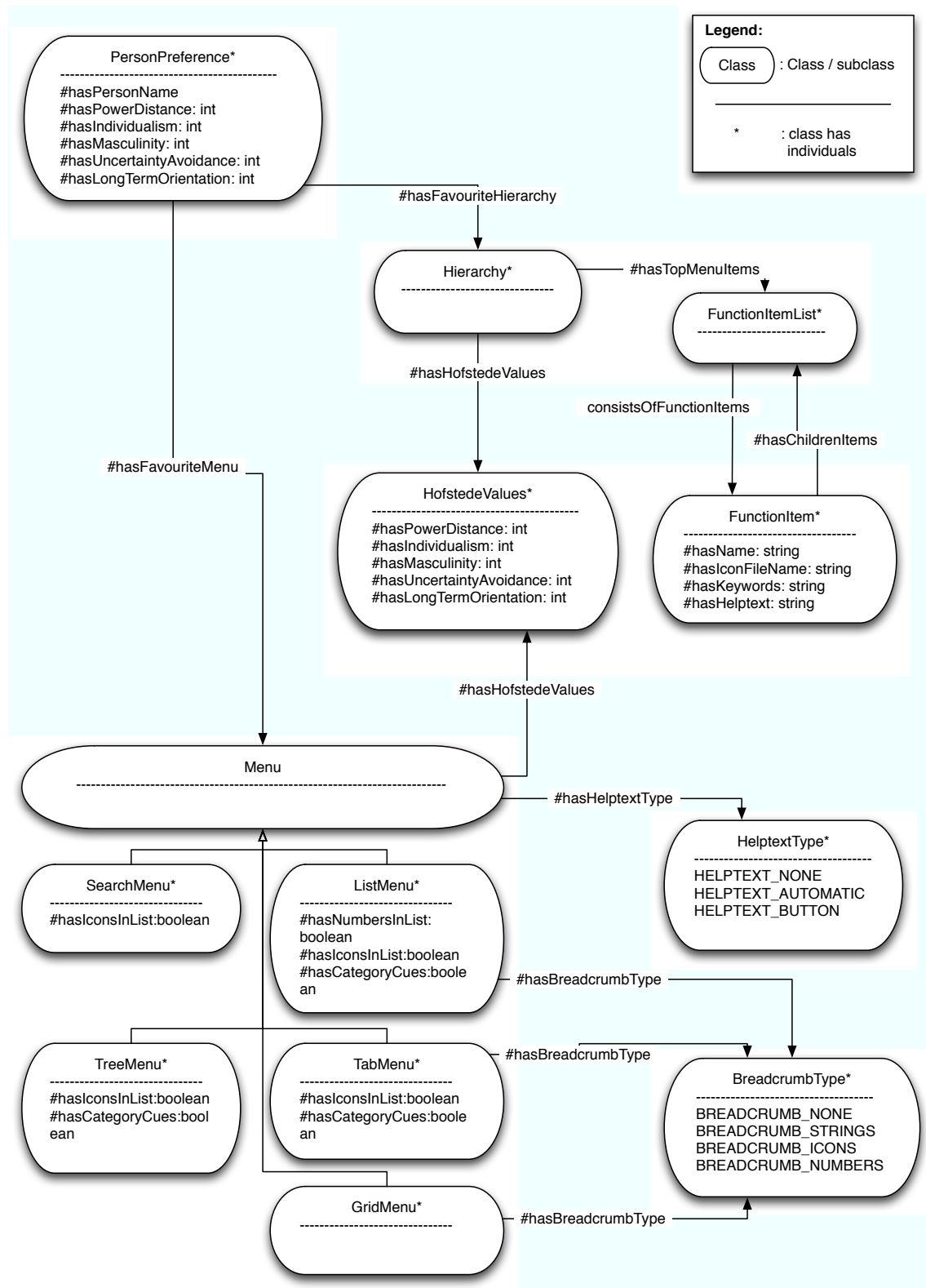


Figure 4.1: The domain model ontology

5

CAMPA - a Culturally Adaptive Mobile Phone Menu Application

5.1 Requirements

The application to be developed is a mobile phone menu. A short questionnaire should ask the user in which countries he has lived. The application then should estimate the cultural background of the user by calculating the Hofstede values of the user based on the information taken from the cultural model ontology CUMO [Reinecke et al., 2007]. Then the application has to adapt its user interface to this background, with regard to number of hierarchy levels, user guidance, the form of navigation and information density. The cultural background is defined by the five cultural dimensions of Hofstede [Hofstede, 2001]. The interface constituents need to be semantically annotated, in order for the application to choose the user interface, that matches the cultural background the best. These semantic annotations are implemented in an adaptation ontology, using the XML-based ontology language OWL. The application should make a connection between the user interface and the two ontologies CUMO and the domain model. Furthermore it has to be able to exchange information with both ontologies. That is, there needs to be some possibility to change the ontologies during runtime.

5.2 Why Java Micro Edition?

5.2.1 The different implementation possibilities

There are different ways to develop an application for mobile devices. It could be developed as a Symbian application programmed in C++, as a Windows Mobile application programmed in C# or Java or as a cell phone application programmed in J2ME; even a web based application would be possible.

Some of these options have some major disadvantages which make it possible to narrow down the list of possibilities. Symbian applications have the advantage that some phone functions can be called from within the application [Nokia Corporation, 2007], but to program the menu application in C++ would be too time-consuming. Furthermore only a restricted number of cell phones support Symbian. The other option that has a major disadvantage is to develop the application as a web based application. Web based applications do not provide the performance needed. After every selection of the user, it takes time to load a new page. This delay prevents fast navigation.

This leads to the following three options left, which were evaluated in more detail:

- Java Micro Edition (J2ME) on a cell phone (CLDC 1.1 & MIDP 2.0)
- Java with the virtual machine IBM J9 on Windows Mobile
- C# with .NET Compact Framework on Windows Mobile

5.2.2 Reasons for choosing Java Micro Edition

Java Micro Edition (J2ME) is pre-installed on most currently sold cell phones. Therefore there are a lot of platforms to choose from and there is no dependency on one platform. The downside of J2ME is that it has almost no capabilities to launch functions of the phone. It only provides possibilities to open an url or dial a phone number.

The two implementations, that rely on Windows Mobile have the advantage of providing access to some functions of the phone via the phone API [Microsoft Corporation, nd; Hage, 2007]. On the other hand they have the disadvantage that the implementation works only on cell phones that have Windows Mobile installed and such cell phones are mostly smartphones like the Palm Treo. These smartphones normally do not look like "normal" cell phones, because they usually have a qwerty-keyboard or a touch-screen. The user experience is not only depending on the design of the application but also on the design of the hardware. Especially for people from third-world countries, who maybe have never seen such a smartphone before, it could look more like a computer to them than a cell phone. In order to make it possible for them to put the navigation-experience of CAMPA in relation to navigation-experiences of their former cell phones, it is necessary to have a similar hardware.

5.3 Evaluation of third-party libraries and frameworks

5.3.1 J2ME GUI libraries

The presentation possibilities of J2ME (MIDP) with high level components are rather limited. There are only containers to display items in a list. Furthermore simple layout changes like changing the background is not possible with the high level user interface API. So there are a lot of user interface libraries that extend the presentation possibilities of J2ME. One of the most sophisticated libraries is J2ME Polish which was chosen as a framework to develop CAMPA.

The most important advantage of J2ME Polish to other GUI libraries is that there are several containers to layout the components of the user interface. Containers like TabbedForm, TreeItem or the possibility to lay out lists as grids. Moreover it is not only a GUI library, it is a whole framework. It gives the developer the possibility to make different builds for different platforms. For instance a release for a Sony Ericsson cell phone makes use of the hardware back button of Sony Ericsson. It even has the possibility to define styles in external files in a CSS-like syntax. So it is possible to separate application logic and layout in a convenient way.

J2ME Polish advantages:

- Sophisticated framework. For example it contains a logging framework and a build framework with which it is easy to compile the application for different cell phone brands and display-sizes.
- Separation between application logic and layout by CSS
- Several layout-possibilities like lists, grids and tabs

5.3.2 XML parser

To get the information out of the ontologies CAMPAs needs some kind of OWL Parser or OWL API. There are Java APIs for OWL like Jena [Jena, nd], Protégé OWL-API [Protégé-OWL API, nd], or The OWL API [The OWL API, nd] but unfortunately these parsers need several classes that J2ME does not have therefore they can not be used for CAMPAs. However, there is a mobile framework that supports ontology processing [Gu et al., 2007], but it was not available when this diploma thesis was started. Moreover it was only tested with an emulator on a P3 500 Mhz processor with 256 MB of memory, which does not prove that it would run on a normal cell phone. For that reason an XML parser was taken and modified so that it was easier to read RDF and OWL files. There are several XML parser libraries for J2ME, and the most important criteria to distinguish between them is how the XML file is parsed. There are three types of parsers: model, push and pull.

A model parser has to generate first a model of the XML-document in the memory before the content of the XML file can be accessed. Mobile devices have very restricted memory capabilities. So it depends on the size of the XML file and the size of the memory of the mobile device whether a model parser can be used.

A push parser is called by the application. Then it parses the whole XML file and looks for certain XML tags, that the application told it to look for. As soon as it finds the tags, it calls listeners in the application. So the parser controls the program flow, which can make the program sequence quite hard to understand.

A Pull parser combines the advantages of model parsers and push parsers. It uses less memory than a model parser but still the control of the program flow is at the application. The application drives the parser through the document by repeatedly requesting the next piece.

One major requirement is that the ontologies can be changed by the program. Changing XML tags is very hard with pull parsers, they are more convenient for generating a new XML file and not for editing. Whereas with a model parser it is easy to edit XML tags in a object-oriented way. Therefore a model parser was chosen: NanoXML for J2ME. Note that the ontologies have to be saved in RDF/XML format, because parsing RDF/XML-ABBREV is too complicated.

5.4 Architecture and program flow

For a rough overview of the program flow see figure 5.1. The Hofstede values of countries are taken from the Cultural User Model Ontology (CUMO) and the Hofstede values of the user interface components are taken from the domain model ontology.

When the application starts it first checks whether there is a domain model saved on the mobile phone from an earlier session containing already created PersonPreferences. It looks for a file called domainmodel.rdf+xml.owl in the folder "/other/" of the mobile phone. If it finds a saved domain model, it loads it, otherwise it loads the domain model given in the jar file. Then the user has the possibility to create a new profile or select the profile, that was saved when he used the application the last time. If he selected a profile the menu of that profile is shown, else the questionnaire to determine the cultural background is started. It asks the user first, which is the country he currently lives in and in the following screens he is asked in which other countries he has lived more than 4 weeks. After the user provided that information, the application takes the Hofstede values of the chosen countries and calculates out of them the Hofstede values of the user. Finally the menu which matches the Hofstede values of the user the best is shown.

5.4.1 Profile selection

The sequence of profile selection is shown in figure 5.2. At the start of the application the domain model is searched for saved profiles. If there are profiles available the user can select one of those profiles in the profile selection screen or decide to create a new profile. If he selects a profile the menu of that profile is shown. If he wants to create a new profile, or has to because there are no profiles available he is directed to a screen where he can enter his name. As soon as he entered his name the questionnaire is opened

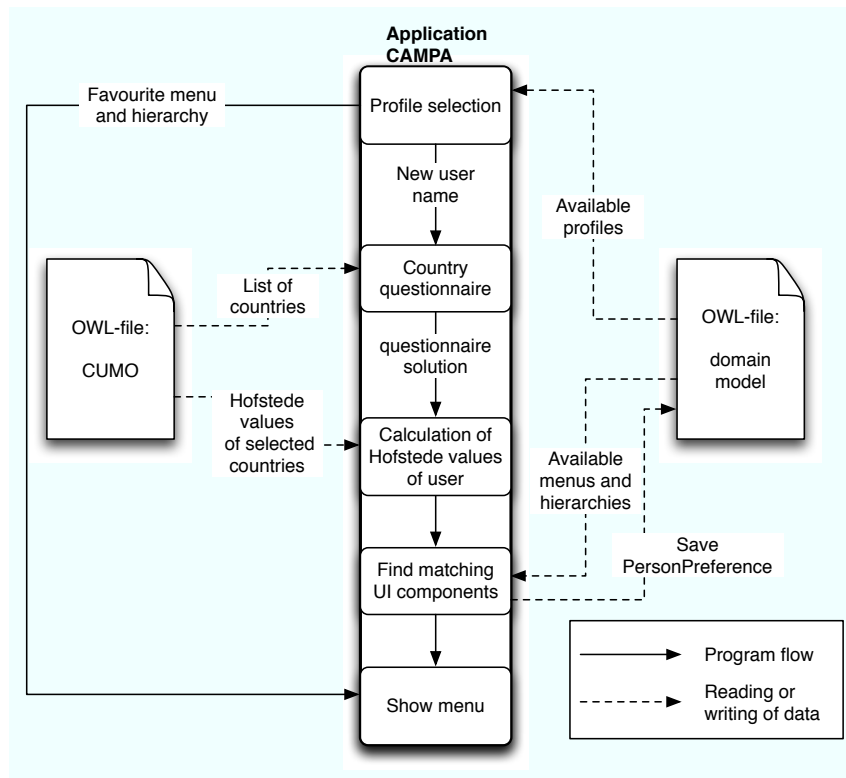


Figure 5.1: Program flow of CAMPA

5.4.2 The country questionnaire

This process is visualized in figure 5.3. The questionnaire starts with a screen, which asks the user in which country he currently lives and how long he has been living there. As soon as the user starts to type the first few characters of a country in the text field, the application suggests some countries in a list which start with the same characters. Those countries are taken from the CUMO ontology. He can only select from countries which are contained in the CUMO ontology. Then the user has to provide how long he has lived in the selected country. He has to provide a number of years or a number of months or both.

After proceeding to the next screen, the user is asked whether he lived in another country for more than four weeks. If he selects "No", the questionnaire is finished. If he selects "Yes", another screen appears, where he can enter how long he lived in that country. This process repeats until the user answers that he has not lived in another country. Then the questionnaire is finished and the application calculates the Hofstede scores for the user.

5.4.3 Calculate the Hofstede values of the user

As soon as the questionnaire has finished the current country and all former countries the user has lived in are determined. The Hofstede values belonging to these countries are then fetched from CUMO. Subsequently the Hofstede values of the user can be calculated. Table 5.1 explains with a pseudo-algorithm how they are calculated. The given example assumes that the user lives currently in Rwanda and formerly lived in South Africa.

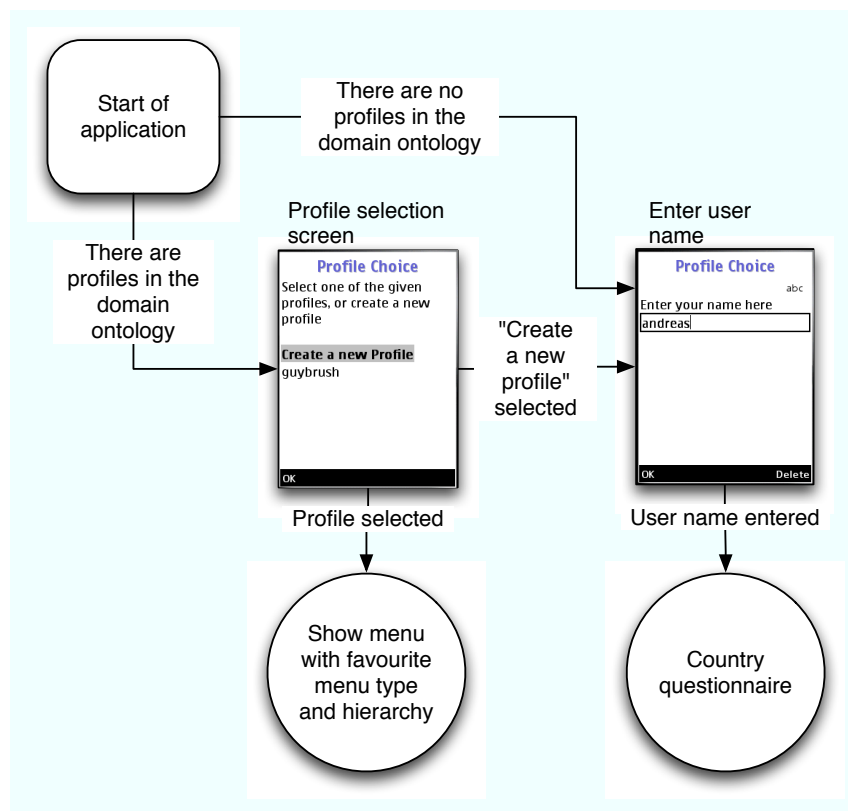


Figure 5.2: The process of the profile selection

Steps	Description	Example
1	Get time spent and Hofstede dimensions of current country and all former countries (LTO/PDI/IDV/UAV/MAS). If one of the dimensions of a country is not defined, take the world average score for that dimension.	Rwanda: 22 Years 25/64/27/52/41 South Africa: 3 years, 4 months -/49/65/49/63 (LTO is undefined, take 45 instead)
2	Calculate the duration of stays in current country and all former countries in months.	Rwanda: 264 months South Africa: 40 months
3	Equate the total of all month durations to 100% and calculate the percental influence of each residence.	100% = 304 months 264/3,04=86,84 Rwanda 40/3,04=13,16 South Africa
4	Take each dimension separately. For each country multiply the dimension with the percentage influence and divide by 100.	Example for PDI: 64*0,86+49*0,13=61,41
5	Round off.	PDI: 61
6	Repeat from step 4 for each dimension.	

Table 5.1: Pseudo-algorithm to calculate the Hofstede Values of a user

5.4.4 Show the menu and hierarchy that match the Hofstede values of the user

After the Hofstede values of the user are known, the menu that matches the cultural background of the user the best can be chosen. For every available menu individual in the domain model the difference between its Hofstede values

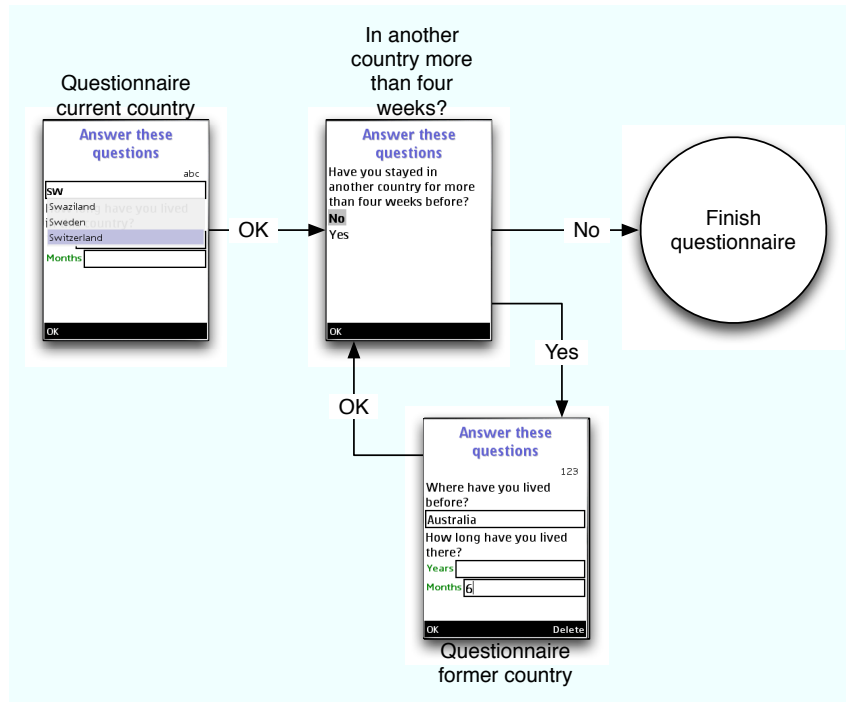


Figure 5.3: The process of the questionnaire which detects current and former countries the user has lived in

and the Hofstede values of the user are calculated and summed. In detail the algorithm works as follows. For each Hofstede dimension the difference is calculated and the absolute value is taken. For example

$$PDI_{difference_menu1} = ABS(PDI_{user} - PDI_{menu1})$$

Then all 5 differences are summed:

$$\begin{aligned} \text{Sum of differences of menu1} = & PDI_{difference_menu1} + LTO_{difference_menu1} + UAV_{difference_menu1} \\ & + IDV_{difference_menu1} + MAS_{difference_menu1} \end{aligned}$$

The lower this sum, the better the Hofstede values of the menu match the Hofstede values of the user. The selection of the best matching hierarchy is done similarly. Note that the selected hierarchy is independent of the selected menu.

As soon as the best matching menu and hierarchy are determined, an instance of *PersonPreference* for the current user is created and added to the domain model. The altered domain model is then saved to the file *domainmodel.rdf.xml.owl* in the folder *"/other/"* of the mobile phone.

5.5 Assignment of Components and Hofstede values

5.5.1 The Menus and components

The following subsections give an overview over the used menu types and components which can enhance the menus. Not every component can be used in each menu. Figure 5.2 outlines which components can be added to which menus. Screenshots of the different menu types can be found in appendix E and the available components are shown in figure 5.4. More explanations about why these menu types and components were chosen are given in section 3.2. During navigation inside the hierarchy the used menu type does not change.

	User interface components										
	No breadcrumb, only function title	Breadcrumb – numbers	Breadcrumb – text	Breadcrumb – icons	No help text	Help text automatic	Help text by button	No icons in lists	Icons in lists	No numbers in list	Numbers in list
ListMenu	x	x	x	x	x	x	x	x	x	x	x
GridMenu	x	x	x	x	x	x	x				
TabMenu	x	x	x	x	x	x	x	x	x		
SearchMenu					x	x		x	x		
TreeMenu					x	x	x	x	x		

Table 5.2: Possible assignments of components to menus

ListMenu

ListMenu orders the menu items vertically in a list. If the user selects a menu item containing sub-items, a new screen opens, which shows the sub-items of the selected category. If the selected menu item has no sub-items, a new screen is opened which tells the user which function he has reached.

GridMenu

GridMenu shows the menu items in a table-like structure. If icons are available they are scaled up and shown with the name of the menu item below the icon, otherwise only the function name is shown.

The chosen grid has three columns. With three columns there is still enough space to display the name of the menu item. Another possibility would have been to display the icons without the name of the icons, and when the user focuses a menu item the name of the item would be displayed in a bar below the title bar. The first option was chosen, because it provides more consistency in the user interface. To illustrate this, take the following example: The menu shows menu items, which all have icons. The icons and the names are displayed. The user selects a menu item. The appearing sub-menu items maybe have no icons, therefore only the names of the functions are shown. With this option the interface is more consistent, because the function names were already shown in the previous menu. With the second option the change is more radical, because in the first menu there are only icons shown and in the second menu there is only text displayed. The navigation style inside a navigation structure with GridMenu is basically the same as in ListMenu.

TabMenu

If available, icons are shown in the tab bar. Otherwise the names of the functions are used. Each tab shows the menu items in a list. That list can have the same properties as ListMenu, except for direct selection by keypad. If the user selects a category function in the list, a new screen appears. The selected function item is focused in the tab bar and the sub-items of that function are shown in the list inside the tab. If the selected menu item is a leaf function which leads directly to a function, a new screen opens that tells the user which function he has reached.

SearchMenu

SearchMenu allows the user to search for all function items that are not categories. That is, all function items that lead directly to a function and contain no sub-function items. Every time the user enters a character in the search textfield, the application searches for function items with name or keywords containing that text and shows them in a list. The name of the function and its keywords are shown. The user can enter the exact name of the function or select from the choices in the list. When he presses the "OK" command a new screen opens that tells him which function he has reached, or an error message if the entered function name could not be found.

TreeMenu

The TreeMenu allows the user to open categories as branches of the tree. Therefore every click on a category menu item only opens a new branch, no new screen is opened. When a leaf function is selected that contains no sub items, a new screen is shown that tells the user which function he has reached.

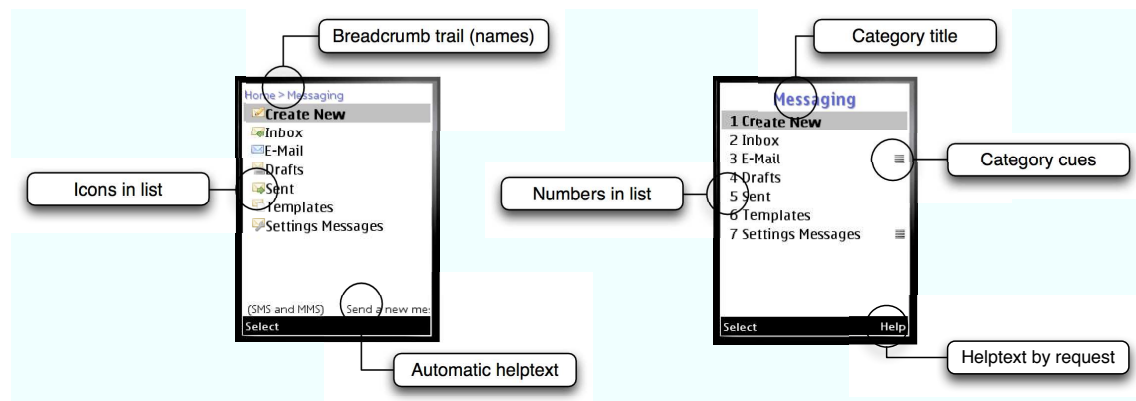


Figure 5.4: All user interface components in two ListMenus

Numbers

This component is only available for ListMenus. If activated, numbers from 1 to 9 are shown between the icon and the function name to indicate that numbers can be directly accessed by pressing the corresponding number key. In the other menu types the direct choice of menu items by keypad was not supported by the framework and would have been too time-consuming to implement.

Icons in lists

Icons can be added as a component in lists in the ListMenu, TabMenu, SearchMenu and TreeMenu. TabMenu (for tabs) and GridMenu (for menu items) inherently show icons. The provided icons have a size of 16 x 16 pixels. 16 pixels is about the height of a menu item in a list.

Category cues

Category cues can be added to ListMenu, TabMenu and TreeMenu. SearchMenu does not show function items that are categories, so category cues make no sense for that menu and in GridMenu there is not enough space to add a category cue to the menu item.

Breadcrumb trails

Breadcrumb trails are available for ListMenu, TabMenu and GridMenu. They are shown in the title bar. If breadcrumb trails are not activated, the title bar shows the name of the current category. SearchMenu does not display multiple hierarchy levels therefore a breadcrumb trail makes no sense. To show the current position inside the hierarchy is already an inherent property of TreeMenu, hence there is no need for a breadcrumb trail.

There are different types of breadcrumb trails (see figure 5.5). The breadcrumb trail can consist of the *names* of the functions, the *icons* of the functions or the *position number* of each function inside each screen. Names are easy to read but can take a lot of space. Icons are not so easy to interpret but do not take much space and the position numbers do not reveal a lot of information about the selected functions, but they can be useful if a user wants to find a function by typing the number combination in ListMenu to have a shortcut to certain functions in the hierarchy that he uses often.



Figure 5.5: The available breadcrumb trails

Help text

Help texts give the user secondary information about functions. There are two possible ways to show the help text. On the one hand a help text can be shown *automatically* by showing a ticker that scrolls the help text at the bottom of the screen as soon as the user focuses a menu item. On the other hand a help text can be shown after the user explicitly requested for help by *pressing a button*. If the user focuses a menu item that has a help text the help command (button) appears. If the user presses the soft key that corresponds to that help command, the help text is shown in a pop-up window. Help texts can be used in every menu except for help text by button which can not be used in SearchMenu because of issues with the framework J2ME Polish.

5.5.2 Assigning certain Hofstede scores to the components

In section 3.2 a rough assignment of Hofstede values to components was done. Depending on the those assignments the following concrete assignments were done, which can be seen in table 5.3.

To have a base to calculate the Hofstede values for a menu consisting of different components, the world averages of Hofstede dimensions was taken, which is 45/55/43/64/50 for LTO/PDI/IDV/UAV/MAS [Hofstede, nd]. First a menu is chosen which adds and subtracts certain Hofstede values from the world average Hofstede scores. For each component that is added (or not added) to the menu, Hofstede scores are added or subtracted from the remaining score.

Menus should have a higher influence on the Hofstede values because they determine the main layout and style of navigation. A difference of 15 was chosen for each affected Hofstede dimension except for GridMenu and SearchMenu which both have only an influence of -5 on the uncertainty avoidance score. GridMenu contains icons, which add more information to the menu items and SearchMenu displays keywords for each menu item which present additional

Menus					User interface components												
	TreeMenu	SearchMenu	TabMenu	GridMenu	ListMenu	No breadcrumb, only function tile	Breadcrumb – numbers	Breadcrumb – text	Breadcrumb – icons	No help text	Help text automatic	Help text by button	No icons in lists	Icons in lists	Numbers with direct-choice	No Cues that flag categories	Cues, which flag categories
LTO		+15				+10	-10	-10	-10								
PDI		+15									+10	-10					
IDV			-15			+10	+10	+10	-10				+10	-10			
UAV		+15	-15	-15		-10	+10	+10	+10	-10	+10	+10	-10	+10	+5	-10	+10
MAS																	

Table 5.3: Change of Hofstede scores depending on used components

information. That plus on information should suit users with a higher uncertainty avoidance [Cha et al., 2005; Choi et al., 2005].

Components should have a weaker impact on the Hofstede values because they take less space on the user interface. Therefore a difference of 10 was chosen for each affected Hofstede dimension. The only exception is the component "Numbers in lists". It has weaker influence on the uncertainty avoidance because it does not provide a lot of information, not as much for example as an icon or a breadcrumb trail.

The algorithm for calculating the Hofstede values of a combined menu is explained in detail in table 5.4. As a simple example a calculation of a GridMenu with help text is shown. Note that the not shown breadcrumb trail also has an influence on the calculation. On the other hand the components "Icons in list", "Numbers in list" and "Cues, which flag categories" are also not shown, but they are not available for GridMenu (according to table 5.2) and therefore are not taken into account for the calculation.

Steps	Description	Example	LTO	PDI	IDV	UAV	MAS
1	Take world average Hofstede score.		45	55	43	64	50
2	Consider the changes, which are caused by the chosen menu type.	GridMenu			-15	-5	
3	Consider the changes, which are caused by the added user interface components.	Help text by button		-10		10	
4	Take into account the components, which are available for this menu, but were not chosen	Breadcrumb trail	10	10		-10	
5	Calculate the Hofstede values for the combined menu	Result	55	55	28	59	50

Table 5.4: Pseudo-algorithm for calculating the Hofstede values of a combined menu

5.6 Hardware requirements

CAMPA was tested on a Sony Ericsson K800i. That phone has a screen-size of 240 x 320 pixels, which is recommended for running CAMPA. The current jar size of CAMPA is about 200 KB. A big part of the jar file size depends on the size of the two contained ontologies: CUMO and the domain model. The allowed jar file size must be taken into account

when selecting a cell phone to run CAMPA on. Currently CAMPA uses the FileConnection API to load and save a domain model from and to the file system of the mobile phone. Therefore the mobile phone should have JSR 75.

6

User evaluation

A qualitative user evaluation of CAMPA was conducted, which is presented in the following sections.

6.1 Objectives

The objectives of the user evaluation were on one side to test some of the assumptions that were made in the guidelines in chapter 3, on the other hand to survey the general usability of the five basic menu types available in the prototype.

6.1.1 Chosen menus for the evaluation

Five menus were composed to evaluate the user interface. Screenshots of them are shown in appendix E. Because of time restrictions, the evaluation was conducted with two main cultural groups in mind. On the one hand Swiss people, of which there are obviously a lot in Switzerland and on the other hand Chinese people. Chinese people were chosen, because almost all their cultural dimensions according to Hofstede are very different to those of Swiss people (see figure 6.1). Based on this decision, firstly two menus were composed, which should suit those two cultures in particular.

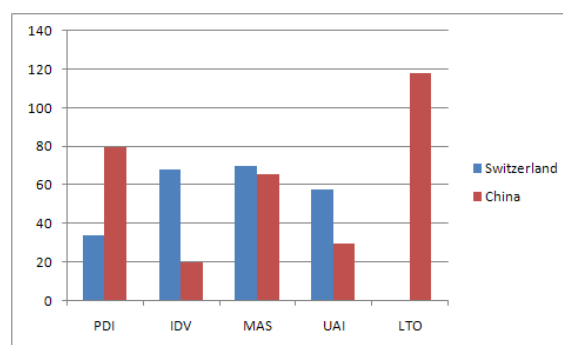


Figure 6.1: Different scores in the Hofstede dimensions for Switzerland and China (from [Hofstede, 2001])

SearchMenu was chosen as a menu for Chinese users because of its high long term orientation score, high power distance index and lower uncertainty avoidance score. An automatic help text was added, to increase the power distance index and icons in the search list for lowering the individualism score.

ListMenu was chosen for Swiss users because most of the other menus change the Hofstede dimensions in an unfavourable way. ListMenu only lowers the uncertainty avoidance score. Moreover ListMenu allows to flexibly add components in order to create a menu that has Hofstede values with a minimal difference to the Hofstede values of Switzerland. In terms of components help text by button was chosen to take into account the low power distance index of Switzerland and no icons to allow for the high individualism. The remaining components "numbers in list", "category cues" and "no breadcrumb trail" were chosen to align the uncertainty avoidance score as much as possible to the uncertainty avoidance score of Switzerland.

After these two menus the following three menu types were used to create menus which have an extreme value in one of their Hofstede scores. GridMenu was used to design a menu with very low individualism score by incorporating an "icon breadcrumb trail". Additionally a "help text by button" was chosen. TabMenu is appropriate for a menu with very high uncertainty avoidance score. To increase the uncertainty avoidance score breadcrumb trail, automatic help text and category cues were added. TreeMenu has no help text, no icons and no category cues in order to keep the uncertainty avoidance score as low as possible.

6.1.2 Chosen navigation hierarchy for the evaluation

The chosen navigation hierarchy is based on the hierarchy of the Sony Ericsson K800i. Some menu items were moved and some were omitted. After the pretest some slight changes were made to the hierarchy. A visual representation of the hierarchy can be found in appendix B.3.

The evaluation could also have been conducted with several different hierarchies, but by keeping the hierarchy structure constant, it is easier to detect the influence of the user interface components.

6.1.3 Hypotheses

Of all the five menus, the composed list menu has the lowest difference to the Hofstede scores of Switzerland and the composed search menu has the lowest difference to the Hofstede values of China. Therefore following two hypotheses are proposed:

H_1 : The composed list menu is the most preferred of the five menus by Swiss participants

H_2 : The composed search menu is the most preferred of the five menus by Chinese participants

6.2 Experimental methodology

The evaluation was done with a Sony Ericsson K800i. Participants were instructed on how to use the "Back button" and the "Clear button" and the navigation joystick which is specific to that phone.

6.2.1 Questionnaires

Two questionnaires were made to get feedback from the user. The task sheet, where the participants rated the different menus and the user questionnaire, where information about the cultural background of the participants and the habits of their cell phone use were recorded. Both questionnaires are available in appendix D.

6.2.2 Evaluation sequence

The participants had to fulfill five tasks and each task in another menu. The conductor of the experiment had a notes sheet (see appendix D) where he entered certain results like the time the participants needed for each task. To avoid learning effects as much as possible the order of the menus was given by following Latin Square [MacKenzie, 2008]:

1	2	3	4	5
2	3	5	1	4
3	5	4	2	1
4	1	2	5	3
5	4	1	3	2

For example the second participant was given menu 2 for task 1, menu 3 for task 2, menu 5 for task 3 et cetera. In the main test special attention was paid that the latin square was used for the Chinese and the Swiss group independently. It was important to make sure that also in the Chinese group each menu was used with each task at least once. Because the application did not lead to an actual function of the phone like writing a text message, there was only a screen for each function that told the users which function they had reached. As soon as the participants thought that they had reached the function that was asked for in the task, they showed the screen of the cell phone to the conductor of the user test, which told them whether they had reached the correct function. If they did not reach the correct function, this would be noted as a mistake on the notes sheet.

After each of the five tasks, the participants rated the menu. They were provided with a screenshot of the menu used, so that they could better remember what it had looked like (the screenshots are provided in appendix E). Firstly they were asked to fill out a form known as the System Usability Scale (SUS) [Brooke, 1996]. Secondly they could write what they liked and disliked about the menu. After completion of all five tasks, the participants were given the opportunity to rate all five menus in relation to each other and to select their most and least favorite menu. In order to facilitate the comparison, screenshots of all five menus were shown.

Finally the participants were asked to fill in the user questionnaire.

6.2.3 User tasks

The participants were asked to fulfill the following tasks. They can also be found in the task sheet in appendix D.

Task 1: You would like to change the ringtone that notifies you when receiving a text message (SMS).

Task 2: You would like to send a text message (SMS).

Task 3: You want to listen to the radio.

Task 4: You have missed several calls. Now you want to see all of these missed calls.

Task 5: You want to turn on the bluetooth.

Some of them are usual (e.g. task 2), some are unusual (e.g. task 1), some in lower levels (e.g. task 5) and some in higher levels (e.g. task 3) of the navigation hierarchy. None of the tasks is in the top level of the navigation hierarchy, so the participants always have to navigate at least into one category to finish the task. For every task the user has to enter a different main category to prevent learning effects. Only task 1 requests the user to enter one of the categories "Messaging" or "Settings" which are also entered in task 2 and task 5.

6.3 Pretest

Firstly a pretest was conducted to eliminate the most obvious mistakes in the tests and in the application. All of the participants of the pretest were Swiss except one participant was German. Eleven participants took part in the pretest.

Their age was from 24 to 30. Three participants were female. Six participants were current or former students of computer science.

6.3.1 Results

An overview of the results of the pretest is given in table 6.1.

	ListMenu	GridMenu	TabMenu	SearchMenu	TreeMenu
Most liked menus:	3	2	4	0	2
Least liked menus:	2	0	1	3	5
Average Time per Menu	16.73	27.91	28.73	69.9	25.91
Average SUS score per Menu	84.55	86.14	80.91	50	65.91
Average rating per Menu	2.27	1.82	1.82	3.55	3.27

	Task1	Task2	Task3	Task4	Task5
Average Time per Task	45.82	20.7	24.27	39.64	34.27
Average SUS per Task	74.55	76.82	77.27	78.41	60.45
Average rating per Task	2.27	2.27	2.36	2.45	3.36

Table 6.1: Overview of the results of the pretest

The ListMenu, GridMenu and TabMenu were rather often rated "most liked". The GridMenu was never rated "least liked". Additionally their average SUS score is high and they received an average rating of approximately 2. The higher the SUS score the better the usability is. The rating scale started at 1 which means "Liked it very much" until 5 which means "Disliked it very much". The average time to solve a task was low for the ListMenu, GridMenu, TabMenu and TreeMenu. The average time to complete a task was the lowest for the ListMenu.

The results for the SearchMenu and TreeMenu are more negative. They were rated more often "least liked" than "most liked". The SearchMenu was never rated "most liked". Both had a lower average SUS score than the other menus and were rated worse on average. Participants needed considerably more time to complete a task in the SearchMenu. Following are some summed up statements of the participants to each menu and to some of the tasks.

ListMenu: Five participants mentioned the good overview as a positive property of this menu. Two users liked the simplicity. Overview and simplicity are also the reasons why this menu was chosen "most liked". The most named negative property was the absence of color or icons.

GridMenu: Nine participants mentioned the icons as a positive property of this menu. This is also the reason for the two selections as "most liked" menu. Three of the answers explicitly said that the icons support the understanding of the functions. Only a few participants mentioned negative statements. One of them mentioned the danger of relying too much on the icons and not reading the text. The icons can be misleading if they do not exactly fit the purpose of the function. Another subject mentioned the change from icons and text to text only. A third participant did not like the inefficient use of the screen space.

TabMenu: Six participants liked the tabs, because they give an overview. Two of those six rated the menu "most liked" because of the good overview. On the other hand seven users mentioned that it can be hard to interpret the icons. Two of these seven thought that there is no good overview, because you have to go through each tab to know the meaning of the tabs. One participant qualified his statement by adding that if he would use the menu frequently he would learn the meaning of the icons.

SearchMenu: Two participants liked the secondary information provided by the keywords. Three participants found it hard to enter text with the keypad. Another three mentioned as a negative point, that they had to know the name of the functions. Two users had no overview because there were too many functions on the screen. One participant mentioned as a disadvantage of the menu that he could not navigate the menu when not looking at it. An advantage of the other menus is that it is possible to access functions by pressing a certain sequence of buttons. So the user does not have to actively look at the screen, but knows that the sequence of "right-up-down" will take him to the function "Send text message".

TreeMenu: Two participants liked that the menu showed the current position inside the hierarchy and another two thought that it gave a good overview. On the other hand three users did not think that it gave good overview and three thought that there was too much information on screen after opening a branch of the tree.

Tasks: Task 1 took the highest average time to complete. This can be due to the fact that this is the only task besides task 5 where the menu item is in the third level of the hierarchy, and therefore the participants need to go deep into the hierarchy, which takes more time. A second reason can be that it was the first task and participants had to first read through all available first level elements. Moreover there are two ways to access this function. One way is through "Messaging" another one is through "Settings" which could be confusing for the participant. One participant first looked for the function in the category "Settings", could not find it there and then looked for it in the category "Messaging".

In task 3 four participants first selected the function "Music player" instead of the function "Radio" which they were asked for. This is due to the position of "Music player" on the first level of the hierarchy and its icon which showed a music note, whereas the function "Radio" was one level deeper in the category "Entertainment". For the main test the function "Music player" will be put inside the category "Entertainment". Interestingly this confusion did not have a big influence on the average time to complete the task. Task 3 was the second fastest task.

6.3.2 Conclusions

Some bugs were found during the pretest in the software and the questionnaires. Therefore the pretest helped to improve the application and the questionnaires. Some participants did not even notice certain components like breadcrumb trail or help texts, so additional questions were added to the notes sheet to check whether the user interface components were noticed and used. Every component has two checkboxes. The first checkbox means "participant noticed the component", the second checkbox means "participant used the component". "Using the component" means that the participant actively had an advantage from the component in some way. For task 1 there are two ways to complete the task, two checkboxes were added to the notes sheet in order to record which path the users took to reach the function. Some users did not select directly the correct function. An additional column keeps track of the number of mistakes the user has made.

The results of the pretest suggest that it is not possible to assign a certain menu type to Switzerland. The composed ListMenu should be the most preferred by Swiss people, but the participants also liked the GridMenu and the TabMenu very much. The participants had different personal preferences. For example some preferred a user interface without icons, others said that they would like icons in the menu. The SearchMenu and the TreeMenu were less liked and also had a lower SUS score. This indicates that the composed SearchMenu and TreeMenu are not suitable for Swiss users. Maybe this finding does not only restrict to Swiss people, but this will be more clear after the main test.

6.4 Main test

The main test was conducted with 18 Swiss and six Chinese participants. One participant was from Liechtenstein and had lived several years in Switzerland. People in Liechtenstein speak also Swiss German therefore he was taken into

the Swiss group. One participant had lived less than half of his life in Switzerland and one participant was Taiwanese. Both were omitted from the evaluation, but can be found in the data sheet on the enclosed CD.

The Swiss groups consisted of eleven male and seven female participants. Eight had a computer science background. 17 were currently students at an university or had been students. Their age ranged from 20 to 33. The Chinese group contained only one female participant. The five male Chinese were doctoral students in computer science. The female participant had a linguistic background. Their age was between 25 and 36.

The two groups represent relatively young users with high education. To get also some knowledge about older users, two test were conducted with two older Swiss female participants. The data can also be found in the data sheet on the enclosed CD and a short overview of the user tests with them is given at the end of this section.

6.4.1 Results

Chinese group

An overview of the results of the Chinese group is given in table 6.2.

	ListMenu	GridMenu	TabMenu	SearchMenu	TreeMenu
Most liked menus:	1	1	3	1	0
Least liked menus:	1	0	0	5	0
Average Time per Menu	52	94.17	51.33	171.67	21.17
Average SUS score per Menu	70.42	75.83	84.58	59.58	76.67
Average rating per Menu	2.17	2	1.83	3.33	2.17

	Task1	Task2	Task3	Task4	Task5
Average Time per Task	117.83	76	48	102.5	46
Average SUS per Task	64.58	81.25	77.08	77.92	66.25
Average rating per Task	2.67	2.17	2.17	2.33	2.17

Table 6.2: Overview of the results of the maintest - Chinese group

The TabMenu was rated "most liked" by half of the group, it had the highest average SUS score and the best rating. On the other hand five participants selected the SearchMenu as "least liked". It had the lowest average SUS score and the worst rating. Moreover the average time to complete a task is highest in this menu, whereas it was lowest for the TreeMenu.

The performance of one participant was well below average. He was the only one, of the Swiss and Chinese group that could not solve all of the tasks. Two tasks were not solved: One Task was the task 1 in the TreeMenu and the second one was Task 5 in the ListMenu. Task 4 took him very long in the GridMenu which explains the high average time for the GridMenu and the task 4.

It was very hard to get information about preferences from the Chinese participants. Most of them were very reserved in providing information about what they liked or disliked about the menus, even after encouraging them to think about what they particularly liked or disliked about the menu. For example one participant could not find the function that was asked for and therefore could not solve the task. But when asked what he did not like about the menu, he said that the menu is fine. After telling him that there must be something negative about the menu because he could not solve the task he said that it was not easy to use and avoided partly to say something negative by stating that he "should ask for help when using this system". This discovery supports the findings of Herman [Herman, 1996] with Asian participants.

The tests with Chinese people took longer than with most Swiss people. Each user test with Swiss people took about 30 to 40 minutes, with Chinese people it typically took about 45 to 60 minutes. This is maybe due to the time they needed in the SearchMenu. Following are some summed up statements of the participants to each menu and to some of the tasks.

ListMenu: One participant mentioned that he prefers icons, another one even would have liked to have animations in the menu. A third one rated the ListMenu "least liked" because it has no icons, but one also rated it "most liked" because he prefers text.

GridMenu: Two participants liked the icons, and a third person rated it "most liked" because of the icons.

TabMenu: Two participants liked the TabMenu because they did not have to enter a category and go back to explore the functionalities, which is also why they rated it "most liked".

SearchMenu: Only one participant used the search function as intended by the designer. He was even fastest in the SearchMenu than in the other menus. He rated it "most liked" because he liked the secondary information provided by the keywords.

The rest of the participants went through the whole list. Some tried to filter by entering text first and gave up, and some directly went through the whole list. One participant mentioned that this could be due to the fact that Chinese characters are hard to enter by keypad. Even though the language of the user interface was English, this maybe has an influence on the willingness to enter text. Accordingly one participant rated it "least liked" because he did not know how to operate it and two gave as a reason, that the menu does not fetch the item that was asked for in the task.

TreeMenu: One participant did not like that it has no icons and another one would even prefer special effects. Interestingly the Chinese group had the tasks solved the fastest in the TreeMenu.

Tasks: Task 1 took longer on average than the other tasks. For one thing this is due to the task, because the participants have to go relatively deep into the hierarchy to solve it, for another thing this can be due to the explanation of the test sequence, which seemed to be not clear for some Chinese participants. Three of them did not realize that they had to find the asked function in the menu. Maybe this should be made more clear in the task sequence description in the task sheet because it was not possible to make sure verbally whether they had understood the task. The Chinese participants did not say that they did not understand what they have to do before the first task started.

One participant gave interesting suggestions how to make the menus better, he did not really focus on the menus but more on the tasks. For task 2 ("Create a new text message") he suggested that it would be more convenient to have a shortcut (could also be provided by a assigned button on the phone) for creating new messages, because the function 'Messaging' "would easily mislead users to read the existing text messages instead of creating a new one. For task 4 he suggested that the menu item "Calls" should have a different color if there are missed calls. Another participant thought there were too many levels for task 5 (3 levels), he would have preferred a shortcut.

Swiss group

An overview of the results of the Swiss group is given in table 6.3.

The GridMenu and the TabMenu were each rated "most liked" by seven participants. So more than two third of the Swiss participants chose one of those two menus as "most liked". Moreover they both had an average rating better than 2, which rates them between "rather liked it" and "liked it very much". Interestingly the participants liked the TabMenu even though the average time to complete a task was highest in this menu.

The ListMenu was rated "least liked" five times even though the average time to complete a task was lowest in this menu. Additionally its average SUS score is rather high. The ListMenu, the TreeMenu and the SearchMenu had a rating of around 3 which means "neither liked it, nor disliked it". The TreeMenu and the SearchMenu were both rated "least liked" six times, but the SearchMenu had a higher average time to complete a task, the lowest average SUS score and the lowest average rating. Following are some summed up statements of the participants to each menu and to some of the tasks.

	ListMenu	GridMenu	TabMenu	SearchMenu	TreeMenu
Most liked menus:	1	7	7	1	2
Least liked menus:	5	0	1	6	6
Average Time per Menu	31.53	48.67	60.22	52.22	42.11
Average SUS score per Menu	83.09	84.17	79.31	62.78	74.72
Average rating per Menu	2.76	1.56	1.83	3.22	2.89

	Task1	Task2	Task3	Task4	Task5
Average Time per Task	91.39	36.5	23.22	26.39	62.17
Average SUS per Task	75.14	82.08	86.11	76.94	62.5
Average rating per Task	2.17	2	2.22	2.72	3.11

Table 6.3: Overview of the results of the maintest - Swiss group

ListMenu: Six participants mentioned that the ListMenu was simple, clear or had no frills and one participant chose it as "most liked" because of its simplicity. On the other hand ten participants thought it needed some icons or color and for five participants this was the reason to chose it as "least liked". The lack of icons had an severe effect for one participant, who still pushed the back button in the top level menu, because he did not know that he was already in the top level menu. He said that the text items looked too much the same for him in all levels of the hierarchy.

The user interface components were also mentioned by some participants, but there was not a trend visible whether they were liked or not. Two participants like the category cues, but another participant did not like them because he did not know what they mean. Two persons liked the numbers for fast access of functions and another one disliked them. The first two had a lot of experience with cell phones and also said that their old Nokia cell phones had such a number feature.

GridMenu: 13 participants (more than two-thirds) explicitly said that they liked the icons in the GridMenu and for seven of them the icons and the color were the reasons to choose this menu as the "most liked". Two of them even would have liked the menu to be more colorful. On the other hand two male participants thought the icons are not so important (more about this topic can be found in the section about gender differences). Six participants did not like that there were no icons anymore in the lower levels of the hierarchy. One participant suggested to show the menu items in a list, if no icons are available, two other participants proposed to have some more space around each item, so it is easier to distinguish between the different items. Two participants favored the two dimensional navigation of the GridMenu over the one dimensional navigation of the ListMenu. For one person the main advantage was that you he did not need so many clicks to select an item and the other person thought it was better for the thumb instead of clicking always up and down. On the other hand another participant did not like the two dimensional navigation because he had more overview on a list and was faster in conceiving the content of the screen. Three subjects did like the breadcrumbs.

TabMenu: Five participants liked the icons or the color in the menu. This was also the reason for four participants to choose it as the "most liked". Two said that it was good to see the current menu they are in which led four participants to choose it as the "most liked". Three subjects mentioned as a positive property that they could navigate on two hierarchy levels at the same time by left-right and up-down and another three liked the tabs because they knew them from internet browsers or their own cell phone.

Seven participants thought that the icons in the tabs are too small. Some of them had problems to understand their meaning and then it was hard to get an overview over all possible menu items because the textual description was only visible if the tab was focused. An implication of the small tabs is that the attention of the user is drawn to the list below the tabs which takes up the most space of the screen, even though the main menu items are in the tabs. This was disliked by four participants. Another problem was that not all the main menu items fitted on the screen, because only six tabs could be displayed, but there were nine main menu items, so the subjects had to scroll to see all possible menu items which was disliked by five of them.

SearchMenu: Six subjects liked the idea of a search menu, amongst other things because they did not have to navigate through a lot of categories and one of them selected it as "most liked". But three thought it was cumbersome to enter text by keypad and eight did not like that they had to know the name of the functions or the keywords to reach the correct function. Three chose it as "least liked" because of this property. Some did not know that the small text below the function name were keywords but thought they were description text. Four liked the additional information given by the keywords. Two participants did not like that there was no browsing possible. Two did not use the SearchMenu as intended and scrolled through the whole list of functions. The help text was liked by one person and disliked by another. Finally three participants thought there is no overview in this menu and chose it "least liked".

TreeMenu: Three participants thought it was intuitive and two liked to see the current position inside the hierarchy. One of them chose it as "most liked" because of this attribute, another one because he thought it was an improved version of the ListMenu. But both of them had tasks which requested them to go only one level deep into the hierarchy. Two subjects liked that it looked similar to applications they had already used like their electronic banking application or Microsoft Outlook. Eight participants missed colors or icons in the menu and four had problems to close the opened branches. These two attributes and the bad overview led five participants to choose the TreeMenu as "least liked". To close the branch the parent node of that branch needed to be selected. Some participants suggested to close the branch by pressing the back button. One participant suggested to close a branch automatically when focusing another parent node. To close the branch by back button was already tried to implement before the pretest, but that would have needed too many changes in the framework of J2ME polish.

Noticed components: Ten participants noticed the breadcrumb trails in the GridMenu or the TabMenu, 13 noticed the numbers in the ListMenu, eleven noticed the automatic help text that scrolled in the SearchMenu and the TabMenu and ten noticed the category cues. The help text button was the least noticed with only seven participants seeing it and not one of them used it. A lot of the participants concentrated fully on the task that they were asked to do, and therefore focused on the menu items.

Tasks: For task 1 it was noted through which categories the participants reached the function "Message Alert", because there were two ways. 11 of 14 participants reached the function by selecting the category "Settings" (4 participants did the task in the SearchMenu, so they did not have to choose a category). Accordingly there was a tendency for Swiss participants to change the message ring tone in the category "Settings" (functional menu structure) and not in the category "Messaging" (thematic menu structure). In future evaluations the desktop-based experiment of Kim et al. could be conducted in a realistic mobile environment. They showed that relational-grouping participants (Korean) were more likely to select and prefer the thematically grouped menu, whereas taxonomic-grouping participants (Dutch) tended to select and prefer the functionally grouped menu.

Task 1 and task 5 took longer than the other tasks. Both require deeper navigation into the navigation hierarchy than the other three tasks. Additionally task 1 is the first task, the participants had to get familiar with the cell phone and the user test. Moreover there are two different paths to solve task 1 and some participants were confused by the names of the functions. Because this also applied to some Chinese participants, this will be discussed further in the conclusions section.

Gender differences between Swiss participants

A main difference between female and male Swiss participants was the desire for color and icons in the user interface. All female participants mentioned that they like the icons in the GridMenu, whereas only about half of the male participants wrote that they like the icons. Almost every female participant explicitly expressed the wish to have more pictures or color in the ListMenu and most of them wanted to have more pictures or color in the TreeMenu, whereas only about a third of the male participants were bothered by too little icons or color in those two menus. One male participant mentioned that he liked the look of the ListMenu because he does not need icons. Another male participant

stated that the icons in the GridMenu are not so important and even disturb him while he is trying to orientate in the menu, therefore he selected the ListMenu as "most liked". Their preferences could also be due to their professional background, because both of them were students of computer science.

Older Swiss users

To see how older users navigate inside the menu a test with two older female participants was conducted. The first one was 58 years old, the second one was 53 years old.

The first subject had very little experience with cell phones. She had a three year old Nokia, and never had a cell phone before. She only sent text messages more than three times a week, all other functions of the cell phone were almost never used. She could only solve two tasks. The second participant was a more experienced user and could solve all 5 tasks. She wrote that she had owned at least four Nokia cell phones.

Both had problems with the small joystick of the Sony Ericsson K800i. They were not used to it and because of their fingernails it was hard to select a menu item because when they tried to push the joystick to select a menu item sometimes the joystick moved in any direction. This was a problem in all of the menus, but they both mentioned that they did not like it in the GridMenu. Their ratings had some similarities with the ratings of the other Swiss participants. For example both chose the TabMenu as "most liked". The first participant liked the SearchMenu also very much, because the menu items were alphabetically ordered.

Bugs found during the main test

In the GridMenu in task 4 it is not possible to click the navigation joystick down to jump from the first to the fourth item which is located in the second row just beneath the first. Left or right has to be used to reach it. This is uncomfortable and was tried to fix even before the pretest. The framework J2ME Polish does some wrong calculations and due to the lack of time it was not possible to implement the needed changes to solve this failure.

In the TabMenu the performance sometimes is not fast enough. When the user selects a menu item the application has to fetch several submenu items from the domain model and this results sometimes in a delay depending on the number of menu items to be fetched. Several people clicked two times, because there was no response from the user interface that the next screen was loading in the background. Two possible improvements could be: Firstly show some information like "Loading...", secondly the loading could be handled better with threads, so that part of the next menu is shown before all items are loaded.

In the SearchMenu when the user does not filter the list of menu items by entering some text, but looks in the whole list of function items, the list is scrolled too fast. The currently focused menu item is not on the screen anymore. The user then sees the seeked function, but can not select it. This was detected because most of the Chinese participants did not enter text but went through the whole list of items.

6.4.2 Limitations

This evaluation can only give information about novice users. A long time evaluation where participants have more time to get used to the menus would maybe show other results. The participants are below 40 years old and almost all of them have an university education. Only very few Chinese people could be found to participate in the evaluation and only one of them was female. Furthermore they almost all have a computer science background. An evaluation with more participants could also take different depths of navigation hierarchies into account.

6.4.3 Conclusions

Validation of the hypotheses:

For Swiss participants the GridMenu and the TabMenu both were most liked and ListMenu was disliked, accordingly there is strong indication that the hypothesis H_1 is wrong. For Chinese participants the SearchMenu was least liked and the TabMenu was most liked. Therefore there is strong indication that the hypothesis H_2 is wrong. The composed TabMenu even has the highest difference to the Hofstede scores of China and Switzerland but still it seems to be preferred by the Chinese and Swiss participants. Nevertheless the average time to complete a task was the lowest in the ListMenu for Swiss participants. Many participants thought that the main negative attribute of this menu was the lack of color or icons. By adding some small icons next to each menu item and some more color perhaps a lot of participants would rate it higher.

These findings indicate that the assignment rules in section 5.5.2 need to be improved and personal preferences may be even more important than the cultural background because the preferences of the individuals within the Swiss group differed. Accordingly the domain model should include other factors besides the Hofstede scores. Very important could be for example the difference between male and female as shown by the differences in desire for icons and color.

Implications for the guidelines

This qualitative evaluation is not really suitable to validate any of the guidelines suggested in section 3.1. Nevertheless following are some comments from the participants that agree or disagree with the guidelines.

Switzerland has a high individualism score and low power distance score compared to the world average scores [Hofstede, nd; Hofstede, 2001]. Even though Switzerland has a high individualism score for which the guidelines recommend to use a low image-to-text ratio a lot of the Swiss participants preferred to have icons in the menus. Three subjects proposed to use color in the TreeMenu to distinguish between branches and higher levels of the navigation hierarchy. This conforms to the guidelines which suggest to use contrast between user interface areas. The guidelines further suggest that users with high power distance would like to be able to initiate communication with the system. Therefore the help text could be opened by the press of a soft key in the ListMenu and the GridMenu. But none of the participants used this help text and only seven saw it. They used the strategy to look in different categories and read the names of the menu items.

China has a high long term orientation and high power distance index compared to the world average. One Chinese subject expressed that he did not like the deep hierarchy in task 5. This conforms to the recommendation in the guidelines to use navigation hierarchies with only a few levels for users with high long term orientation and high power distance index.

How to conduct a user test with Chinese people:

The learned lessons from conducting user tests with Chinese people are shown here. The main problem was that most of them were very reserved when asked what they liked and disliked about the menu even though they were asked to speak freely about their likes and dislikes concerning the menu. Still it seemed like they did not want to criticise the menus but encouraging them verbally helped to get more feedback as suggested by Herman [Herman, 1996]. So this suggestion by Herman proved to be useful. Following there are some more suggestions how to get more feedback from Chinese users.

First of all, questions for Chinese people should be formulated different than for Swiss people. One possibility would be not to ask them about what they liked or disliked about the menu, but to ask for possible improvements. Because the first question style forces the Chinese to criticise the system whereas the second question style gives them the possibility to help the questioner.

Another possibility would be to ask more detailed questions about specific properties of interfaces. Examples for such questions could be: "Do you prefer colored interfaces?", "Do you prefer interfaces with icons to text-only interfaces?" or "Should an interface show your position inside the hierarchy?".

Insights concerning the different menus and tasks

ListMenu: ListMenu can be used on all levels of the navigation hierarchy, because it does not depend on whether icons are available or not. Nevertheless a lot of participants wished to have icons or color in the ListMenu. One participant had problems to navigate, because there were no icons. Accordingly the menu items in ListMenu should have icons if they are categories, so the user can remember the menu better if he returns from a deeper level of the hierarchy. If all the items in a menu lead directly to a function and their meaning is too similar to be represented by different icons, they can be shown without icons.

GridMenu: In some levels of the navigation hierarchy the functions did not have icons. When displaying such a level of the hierarchy in GridMenu the menu items are squeezed together, so it is hard to distinguish between the different menu items. More space between the items could improve the readability, but one participant suggested to display them as a list, if no icons are available. The advantage of the list presentation is that long function names are not wrapped. Grid menus should be only second choice if there are no icons.

TabMenu: Some participants did not like to scroll through the tabs to reach for example the category "Settings". This indicates that tab menus should only be used for levels of the hierarchy which contain only as much items as fit as tabs in one screen. Another disadvantage of tab menus is that it is not possible to see all function names of the tabs at a glance. So this is another reason why there should be as few tabs as possible.

SearchMenu: The SearchMenu was disliked by five of the Chinese participants, so it seems the antipathy towards this menu does not only hold true for Swiss subjects as assumed after the pretest and the main test with Swiss people. This could be due to a general bad usability of this menu or because it may be harder to conceive function names, than to read through categories and narrow the possible functions. These two factors are discussed further below. But perhaps it is also disliked because people are unused to enter text to search for a function. This could especially apply to Chinese users because to enter Chinese characters is very cumbersome on a keypad and this could also affect their willingness to enter English text.

Several participants did not like that they had to know the name of the function in order to find it and many of them thought the keywords were some description text. This implies multiple conclusions. The functions need to be well tagged with keywords in order to make it as easy as possible for the user to find the desired function. To give the user a hint about how the search works and that there are keywords which are also searched, the currently entered text could be highlighted in the keywords or the function name. For example if the user entered "mes" and one of the keywords of the found function is "Messaging" the part "Mes" could be highlighted.

In further evaluations it should be considered to use other names for the functions in the SearchMenu than in other menu types. "My Devices" may be meaningful when browsing through the category "Bluetooth", but not if you just searched for the function to turn on or off Bluetooth.

Some subjects missed the possibility to browse in the SearchMenu. This maybe relates to the problem of not knowing the name of the functions. Because to remember or to conceive function names needs the active participation of the user, whereas a hierarchical navigation based on categories presents the the user some main categories he can choose from and reduces the set of possible menu items with each choice. The user does not have to conceive function names, but can passively process the given options. Moreover browsing is essential to find out what functions the cell phone provides. Therefore the SearchMenu should not be the only menu to access functions, but for some users it would be a good secondary menu for functions which are not used very often.

TreeMenu: Interestingly the average time to complete a task was the lowest in the TreeMenu for Chinese participants. Also none of them chose this menu "least liked" as opposed to Swiss subjects in the pretest and the main test. An evaluation with more Chinese participants could assess if this menu is more suitable for Chinese people than for Swiss people.

Scrolling down was mentioned as a negative point by several people. This is especially a problem after opening a branch at the bottom of the screen. After opening a branch the menu should maybe scroll automatically down to show more of the branch.

The back button should close the currently opened branch to give the user a consistent way to navigate in the menu. When the user is inside a branch and focuses another parent node the currently opened branch should close. Moreover several participants had problems to distinguish between the opened branches and higher levels of the hierarchy. Color could help to improve the contrast between the hierarchy levels.

Task 1: Several Swiss and Chinese participants selected in task 1 the function "Ring tone" instead of "Message Alarm". This could only happen, if the participant chose the "Settings" category. Maybe the function "Message Alert" should be renamed. Another solution would be to change the task description, because it used the keyword "ring tone".

Task 5: Another common mistake by Swiss and Chinese subjects occurred in task 5. They looked for the Bluetooth function in the categories "Organizer & Tools", "Entertainment" or "My Files" instead of "Settings". One participant said that he understands "Settings" as preferences of the phone that are running in the background like time settings for example. Bluetooth was for him not such a preference but a communication tool and therefore should be in the category "Tools". A Chinese participant said, that he would like for example to send pictures to other phones that is why he selected "My Files" in the first place. He suggested to make it possible to first select the picture, then select how to send it (by Bluetooth or by infrared) and then if the selected device is not turned on, the system should turn it on and send the picture. So there would be no need to explicitly turn on Bluetooth.

Maybe the Bluetooth function should be replaced or there should be two ways to access it, through "Settings" or through "Organizer & Tools". It also has to be taken into concern that some subjects did not know what Bluetooth is. The explanation of Bluetooth by the conductor can then influence where they look for the Bluetooth function. Maybe another function should be used for task 5.

Methodology of the user test

In future evaluations screenshots with dummy menu items should be used, so that participants do not get too much knowledge about the hierarchy structure. One participant saw the Bluetooth icon on a screenshot of task 4, had a look at the breadcrumb trail and then knew where to find the Bluetooth function for task 5.

When doing evaluations with different cultures, the test sequence should be exactly the same. For example in the current evaluations some verbal explanations were given to the participants before the user test was started to make sure they understood the test sequence. The Swiss group received them in Swiss German and the Chinese group in English. In a future evaluation it would be better if the explanations would be given in English to all the participants.

Habits

Some people liked some components of the menus because they were similar to already known interfaces. For example the TabMenu reminded them of the tabs in their internet browser or of the tab menu of their cell phone. The SearchMenu reminded one person of iTunes because it filtered the list of functions with every entered character. Being already used to some sort of navigation can have a big influence on the rating of navigation menus. Only two of the seven Swiss participants who chose the TabMenu as "most liked" have ever use a tab menu on a cell phone, but all Swiss subjects who selected the GridMenu as "most liked" have already used grid menus on a cell phone. On the other hand the three Chinese users who chose the TabMenu as "most liked" have already used tab menus on a cell phone. So when looking at the affection of Swiss users towards the GridMenu and of Chinese users towards this finding should be kept in mind.

The role of icons and color

Icons and color not only make the application more appealing to some users but they also act as information about the current position inside the hierarchy. In the ListMenu one participant did not recognize the top level menu and continued to push the back button because all the text lists looked the same to him.

In the TreeMenu colors could be used to give a contrast between super-menu items and submenu items. The advantages of using icons mentioned by participants were that it is easier to find out the meaning of a menu item and to recognize already used menu items. On the other hand colors and icons can influence the readability. A few users liked the clean look of the ListMenu and preferred to read text instead of icons.

Older users

The user test with the older Swiss participants showed that for an evaluation with older people a cell phone without joystick should be used. Both of them had problems to navigate in the menu and select items because of the joystick.

7

Limitations and Future Work

The presentation of information (see section 1.2.3) is not adapted to the culture of the user. For example the color the background or the writing direction do not change. This information is country specific and therefore the data about presentation preferences of countries would need to be collected which is beyond the scope of this thesis. Furthermore the current available user interface components that can be combined should make it possible to test the adaptivity algorithms, which was the cause for implementing the prototype.

Reading direction: Different cultures have different reading and writing directions [Marcus et al., 1999]. Research in this topic indicates, that there is no homogeneous reading direction in China. Chan and Bergen say that Chinese speakers represent a left-to-right writing system. In their experiment English and Chinese speakers focused on the left first. Taiwanese speakers focused on the right first. This is because most current Taiwanese writing runs top-to-bottom starting on the right, small amounts of current literature are beginning to adopt the English-like left-to-right style [Chan and Bergen, 2005]. Chun et al. say that common Chinese publications like newspapers, textbooks use predominantly horizontal Z orientation from left to right, while in Hong Kong vertical N orientation from top to bottom predominates [Chung et al., 2000]. Goonetilleke et al. showed, that Hong Kong Chinese use predominantly horizontal search patterns while the Mainland Chinese change their search pattern depending on the layout presented [Goonetilleke et al., 2002]. This indicates, that it would be quite hard to assign a reading direction to certain countries like China.

Furthermore researchers do not agree on how important the adaptation of the user interface towards the writing direction of the user is. The experimental results of Shih and Goonetilleke indicate that for the Chinese population, a horizontal menu in either language is more effective than the vertical orientation [Shih and Goonetilleke, 1998]. On the other side Röse recommends to take into account the vertical reading direction for the Chinese market when positioning interaction- and information-components on the user interface. This recommendation is based on an experiment with Chinese characters. Consistent with this finding are the results of Dong and Salvendy, which indicate that horizontal English menus result in better performance than vertical English menus. However, vertical Chinese menus result in better performance than horizontal Chinese menus [Dong and Salvendy, 1999]. This indicates that implementing the left to right reading direction would only make sense, if also other languages are supported by the user interface, like for example Chinese characters, which is not the case in CAMPA.

Finally the framework of J2ME polish does not provide an easy way to change the layout from left-to-right to right-to-left. All that research combined supports the omission of adaptation of the reading direction for user interfaces with Latin characters.

Culture specific cell phone functionality: Certain cultures use certain functionality of their cell phone more often than other cultures. For example "beeping" is a common phenomenon in certain African countries [Donner, 2007]. The application could propose preferred functions and for example display them directly in the top menu list.

This was not implemented because the information about preferred functions is country specific. Therefore data would be needed to define which countries use specific functions. Unfortunately such data is rare. More research is necessary in order to implement this function.

Localized icons or colors: CAMPA does not incorporate localized icons or colors. Colors can have different meanings in different cultures [Dreyfuss, 1984]. For localizing icons for regions or even countries, different factors have to be taken into account, like meaning of colors [Dreyfuss, 1984], symbolic meaning of icons [Fernandes, 1995] or level of abstraction of the icons [Kim and Lee, 2005; Röse, 2005b]. These factors would need to be evaluated first for each region or country with help of people from these regions.

Contrast of different user interface areas: In the guidelines in section 3.1 there is the recommendation to use contrast between interface areas for user with high individualism score. In order to use contrast, different colors would be needed. Again, the different meanings of the colors used for each country would need to be evaluated first [Dreyfuss, 1984]. Otherwise the used colors could influence the perception of the user interface in negative ways.

Multiple languages: CAMPA is currently available in English only. To translate every menu item into another language would be beyond the scope of this thesis. Even though it would be desirable to use for example Chinese characters to test the change of the reading direction with Chinese users.

Assignment of menu types to certain levels of the navigation hierarchy: The navigation hierarchies could be defined differently in the domain model in order to make it possible to assign different menus to each level of the hierarchy. This could be useful because some menu types are more useful in the top levels of the hierarchy and some are more suitable for lower levels of the hierarchy.

Further adaptations during use of the application: An editor could be implemented to give the user the possibility to edit his personal preferences and his Hofstede values after the first adaptation. The user interface should then change based on these new preferences and Hofstede scores. An big challenge would be to log the interaction of the user with the interface and automatically adapt the presentation of information and the hierarchy structure. For example regularly used objects would then become shortcuts in the top level of the hierarchy. This would have to be tested in a long term evaluation.

Extension of the domain model with personal properties: The domain model could be extended with more properties of the user, not only cultural properties. As the evaluation showed it could be already helpful to know the gender of the user.

8

Conclusions

A culturally adaptive mobile phone application (CAMPa) was implemented in J2ME that calculates the cultural background of a user depending on the Hofstede scores of the countries he has lived in and presents him an mobile phone menu navigation which should suit him based on his cultural background. A menu consists of a of a menu type, different components and a navigation hierarchy. The menus and hierarchies and their assignment to Hofstede scores need to be defined in a domain model ontology.

CAMPa provides five different menu types (a list menu, a grid menu, a tab menu, a search menu and a tree menu) and has different user interface components to change the presentation of these menu types: Icons, different breadcrumb trails, numbers in the list menu, icons to mark a category which contains items, automatic help text scrolling in a ticker, help text that appears by the push of a soft key.

The conducted user tests with Swiss and Chinese participants showed that CAMPa can be used to evaluate navigation preferences in mobile phone menus. Unfortunately this qualitative evaluation was not comprehensive enough to validate the suggested concrete guidelines. In order to accomplish this an evaluation with more cultures and more subjects would have to be done. However, some comments from the participants indicate both, agreement and disagreement with the guidelines. Furthermore the results of the evaluation indicate that the assignment of Hofstede scores to the components of CAMPa as suggested in this thesis may be wrong. This shows that the assignment needs to be improved in order to make it possible to present the user a menu navigation which suits his cultural background. The results also question the concept of an adaptive menu navigation based solely on the information in what countries the user has lived in. It is suggested to incorporate other personal information into the domain model besides the calculated Hofstede scores of the user. Furthermore the evaluation gave interesting insights what the advantages and disadvantages of the composed navigation menus are and what to keep in mind when conducting cross cultural evaluations.

A

Appendix: Definition of income groups

These definitions are taken from [ITU, 2006].

A.1 Low income

Low income group aggregate. Low-income economies are those in which 2004 GNI per capita was \$825 or less. The economies included are:

Afghanistan	Korea, Dem. Rep.	Solomon Islands
Bangladesh	Kyrgyz Republic	Somalia
Benin	Lao PDR	Sudan
Bhutan	Lesotho	Tajikistan
Burkina Faso	Liberia	Tanzania
Burundi	Madagascar	Timor-Leste
Cambodia	Malawi	Togo
Cameroon	Mali	Uganda
Central African Republic	Mauritania	Uzbekistan
Chad	Moldova	Vietnam
Comoros	Mongolia	Yemen, Rep.
Congo, Dem. Rep.	Mozambique	Zambia
Congo, Rep.	Myanmar	Zimbabwe
Cote d'Ivoire	Nepal	
Eritrea	Nicaragua	
Ethiopia	Niger	
Gambia, The	Nigeria	
Ghana	Pakistan	
Guinea	Papua New Guinea	
Guinea-Bissau	Rwanda	
Haiti	Sao Tome and Principe	
India	Senegal	
Kenya	Sierra Leone	

A.2 Middle income

Middle income group aggregate. Middle-income economies are those in which 2004 GNI per capita was between \$825 and \$10,065. The economies included are:

Albania	Fiji	Panama
Algeria	Gabon	Paraguay
American Samoa	Georgia	Peru
Angola	Grenada	Philippines
Antigua and Barbuda	Guatemala	Poland
Argentina	Guyana	Romania
Armenia	Honduras	Russian Federation
Azerbaijan	Hungary	Samoa
Barbados	Indonesia	Serbia and Montenegro
Belarus	Iran, Islamic Rep.	Seychelles
Belize	Iraq	Slovak Republic
Bolivia	Jamaica	South Africa
Bosnia and Herzegovina	Jordan	Sri Lanka
Botswana	Kazakhstan	St. Kitts and Nevis
Brazil	Kiribati	St. Lucia
Bulgaria	Latvia	St. Vincent and the Grenadines
Cape Verde	Lebanon	Suriname
Chile	Libya	Swaziland
China	Lithuania	Syrian Arab Republic
Colombia	Macedonia, FYR	Thailand
Costa Rica	Malaysia	Tonga
Croatia	Maldives	Trinidad and Tobago
Cuba	Marshall Islands	Tunisia
Czech Republic	Mauritius	Turkey
Djibouti	Mayotte	Turkmenistan
Dominica	Mexico	Ukraine
Dominican Republic	Micronesia, Fed. Sts.	Uruguay
Ecuador	Morocco	Vanuatu
Egypt, Arab Rep.	Namibia	Venezuela, RB
El Salvador	Northern Mariana Islands	West Bank and Gaza
Equatorial Guinea	Oman	
Estonia	Palau	

A.3 High income

High income group aggregate. High-income economies are those in which 2004 GNI per capita was \$10,066 or more. The economies included (plus Taiwan, China) are:

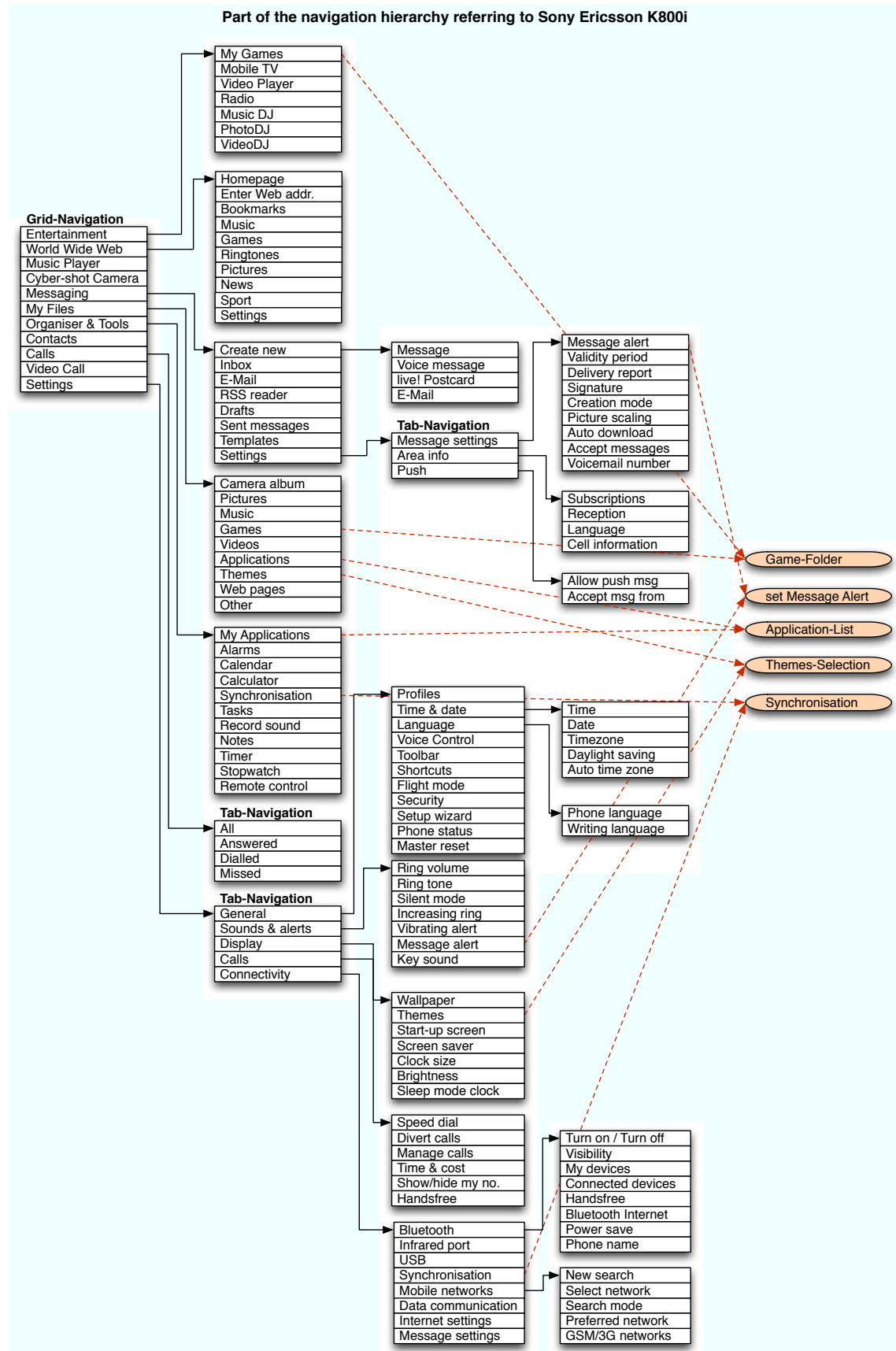
Andorra	Greenland	Norway
Aruba	Guam	Portugal
Australia	Hong Kong, China	Puerto Rico
Austria	Iceland	Qatar
Bahamas, The	Ireland	San Marino
Bahrain	Isle of Man	Saudi Arabia
Belgium	Israel	Singapore
Bermuda	Italy	Slovenia
Brunei	Japan	Spain
Canada	Korea, Rep.	Sweden
Cayman Islands	Kuwait	Switzerland
Channel Islands	Liechtenstein	United Arab Emirates
Cyprus	Luxembourg	United Kingdom
Denmark	Macao, China	United States
Faeroe Islands	Malta	Virgin Islands (U.S.)
Finland	Monaco	
France	Netherlands	
French Polynesia	Netherlands Antilles	
Germany	New Caledonia	
Greece	New Zealand	

B

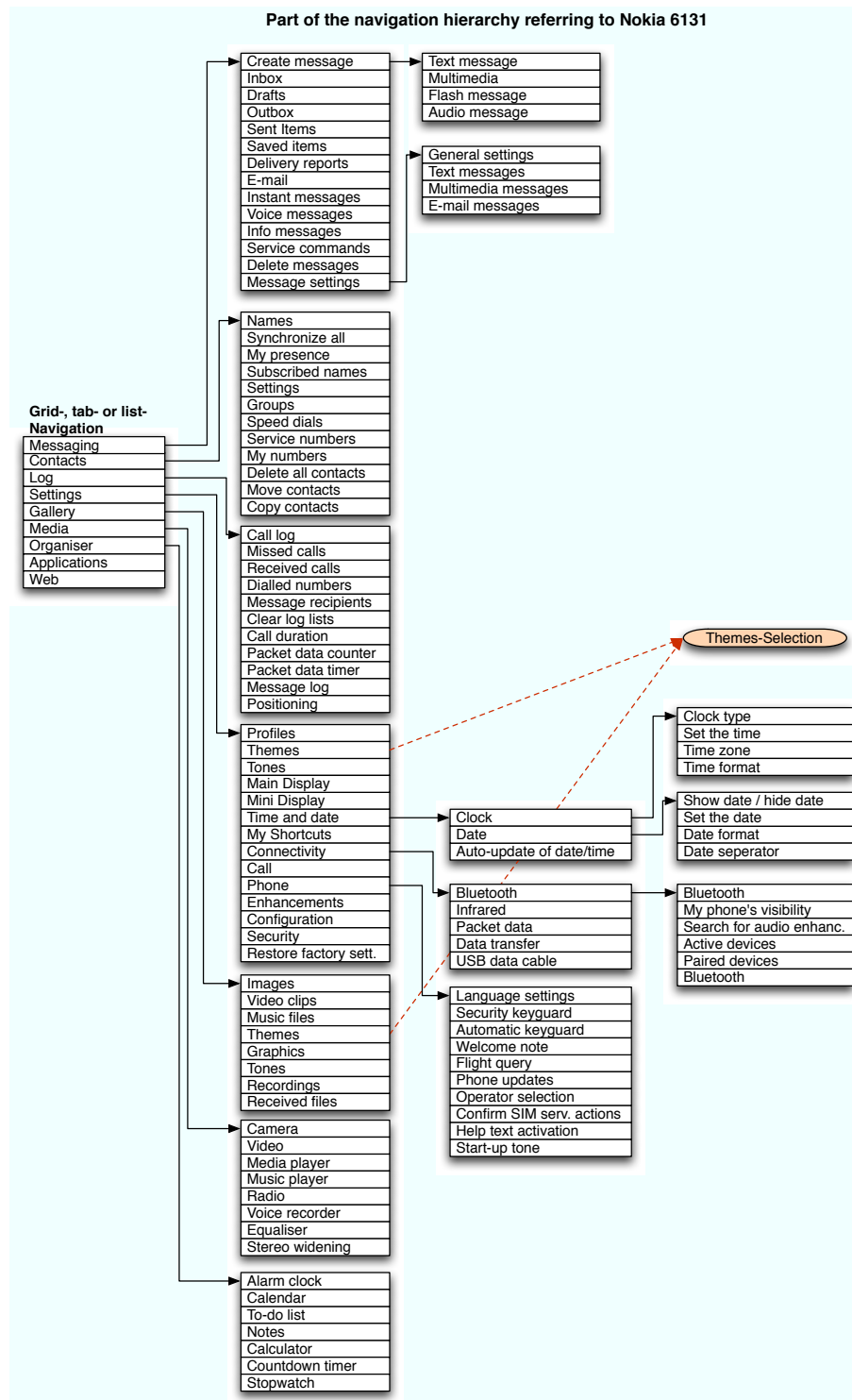
Appendix: Navigation hierarchies of evaluated cell phones and CAMPA

Only parts of the navigation hierarchies were evaluated, because to evaluate every detail of the hierarchy structure would be too extensive.

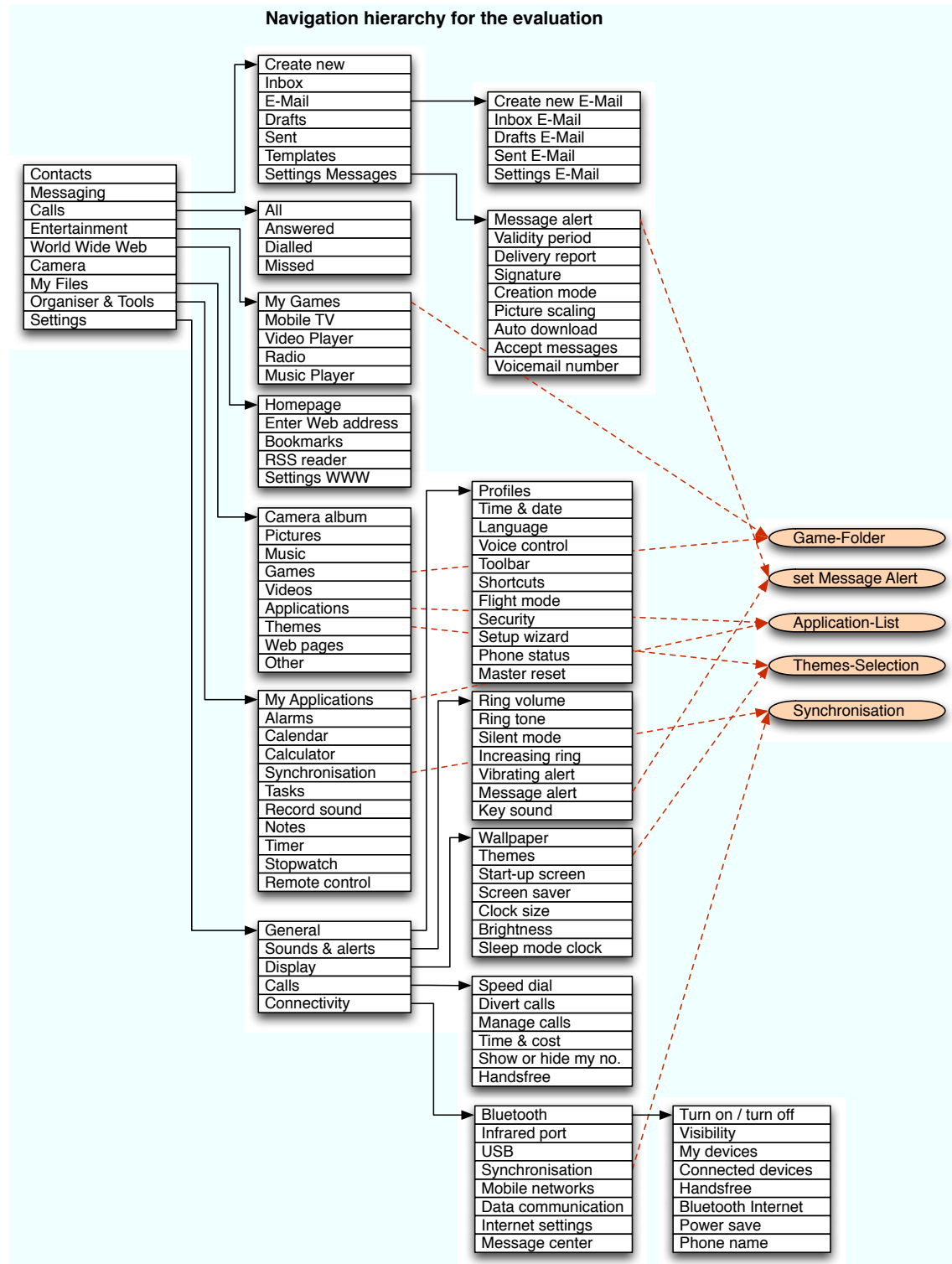
B.1 Hierarchy structure of Sony Ericsson K800i



B.2 Hierarchy structure of Nokia 6131



B.3 Hierarchy structure of CAMPA for evaluation



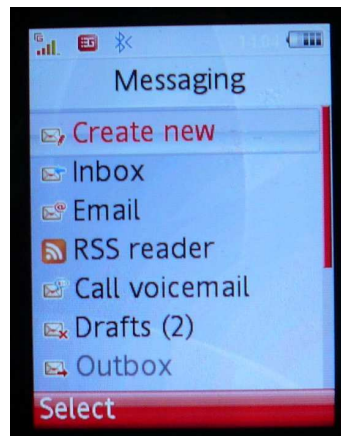
C

Appendix: Screenshots of cell phones

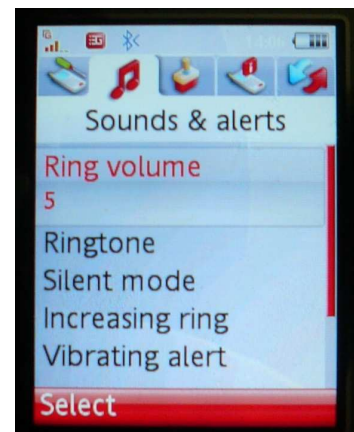
C.1 All screenshots of Sony Ericsson K800i



(a) Main menu



(b) Messaging menu

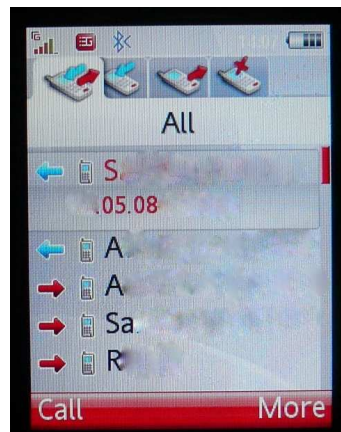


(c) Settings menu

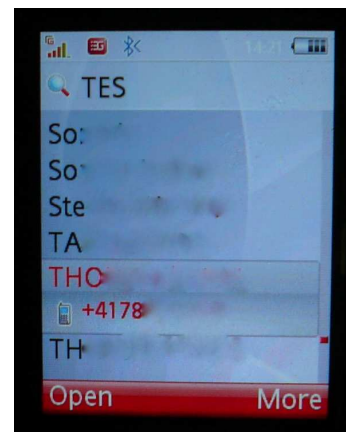
Figure C.1: All screenshots of Sony Ericsson K800i



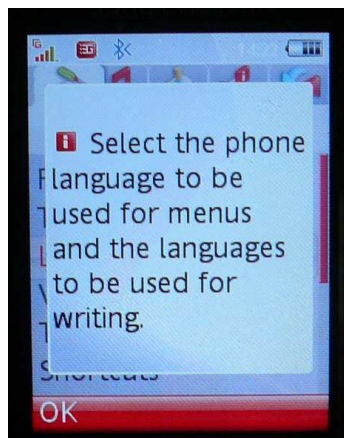
(d) Main menu



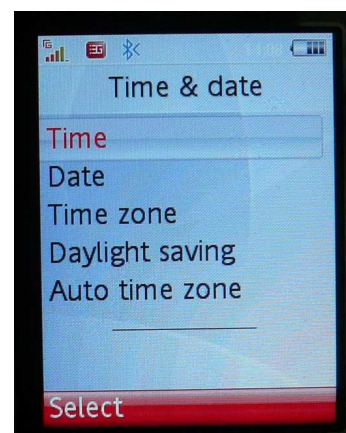
(e) Calls menu



(f) Contacts fast search



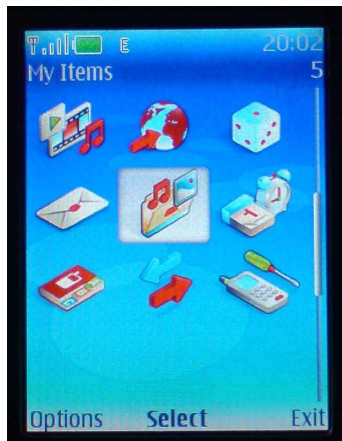
(g) Help text pop up



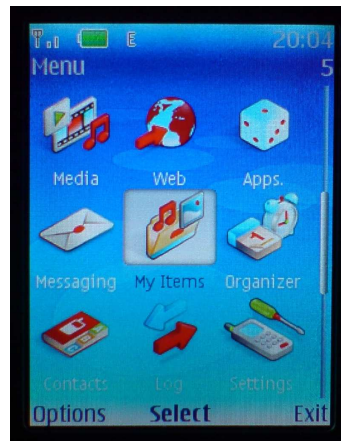
(h) Time and date settings

Figure C.1: All screenshots of Sony Ericsson K800i (cont.)

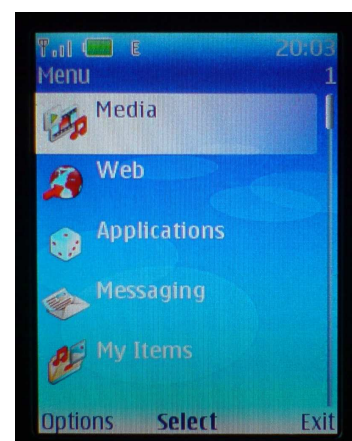
C.2 All screenshots of Nokia 6131



(a) Main menu as grid menu with icons only



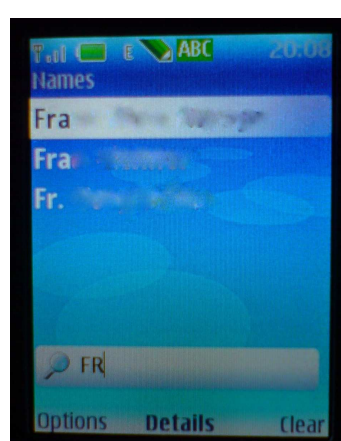
(b) Main menu as grid with icons and text



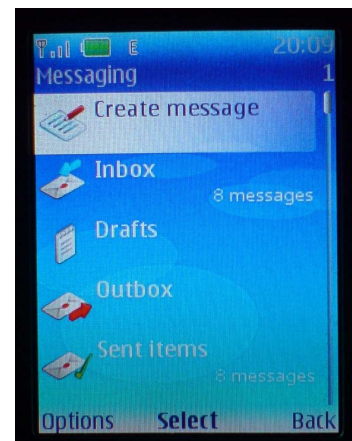
(c) Main menu as list menu



(d) Main menu as tab menu

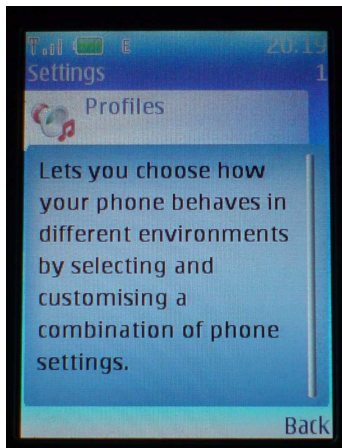


(e) Contacts filter search

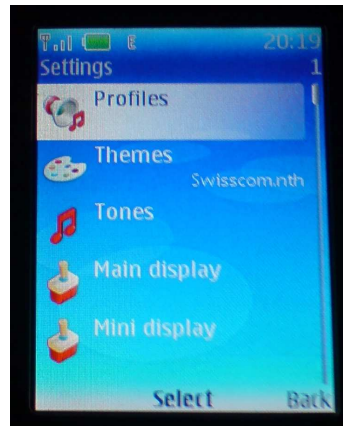


(f) Messaging menu

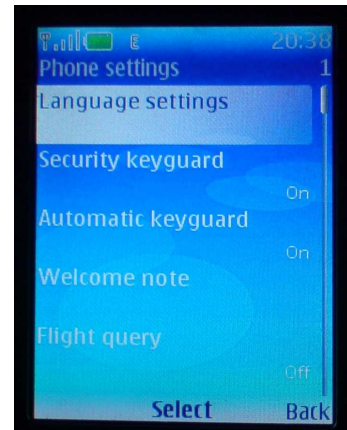
Figure C.2: All screenshots of Nokia 6131



(g) Help text pop up



(h) Settings menu, list menu with icons



(i) Phone settings menu, list menu without icons

Figure C.2: All screenshots of Nokia 6131 (cont.)

C.3 Screenshots of DoCoMo FOMA N905i



(a) Phonebook



(b) Phonebook contact details



Alphabet



Memory No.



Group

(c) Tabs in the phonebook

Figure C.3: Screenshots of DoCoMo FOMA N905i (from [NTT DoCoMo, 2007])

D

Appendix: Forms for the user evaluation

Following forms were used for the evaluation of CAMPA. The notes sheet was filled out by the conductor of the user test, the Task sheet and the User questionnaire was filled out by the participant. They are further explained in section 6.2.

D.1 Notes sheet

Evaluation on cultural preferences in mobile phone interfaces

Notes sheet

Date:

	Menu	Start (min,sec)	-	End (min,sec)	=	Time (sec)	Mistakes
Task 1	<input type="text"/>	<input type="text"/>	-	<input type="text"/>	=	<input type="text"/>	<input type="text"/>
Task 2	<input type="text"/>	<input type="text"/>	-	<input type="text"/>	=	<input type="text"/>	<input type="text"/>
Task 3	<input type="text"/>	<input type="text"/>	-	<input type="text"/>	=	<input type="text"/>	<input type="text"/>
Task 4	<input type="text"/>	<input type="text"/>	-	<input type="text"/>	=	<input type="text"/>	<input type="text"/>
Task 5	<input type="text"/>	<input type="text"/>	-	<input type="text"/>	=	<input type="text"/>	<input type="text"/>

If a task can be reached in two ways, ask the participant, which way he chose.

Task 1: How was the function Message Alert reached? Through

☐ Messaging

☐ Settings

Did participant notice one of the following components?

☐ Breadcrumb trails

☐ Helptext automatic

☐ Category cues

☐ Numbers in lists

☐ Helptext by button

D.2 Task sheet

Evaluation on cultural preferences in mobile phone interfaces

Tasksheet

The intention of this study is to analyze the connection between the navigational preferences in mobile applications, and the users cultural background.

Any data from this survey will be treated anonymously.

Test-Sequence

You will be asked to fulfil the following tasks in different navigational views of a mobile phone menu. Please be aware that this is just a test application, that is, the phone's functionality is switched off. You will see a screen telling you which functionality you have reached. („You have reached the function 'FunctionName'")

Please let me know if you think you have reached the function that I asked for in the following tasks.

Task 1

You would like to change the ringtone that notifies you when receiving a text message (SMS).

Questions for Task 1

If you feel that you cannot respond to a particular item, mark the center point of the scale

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system.	1	2	3	4	5

What did you particularly like about this menu (answers also possible in German)?

What did you dislike about this menu?

Task 2

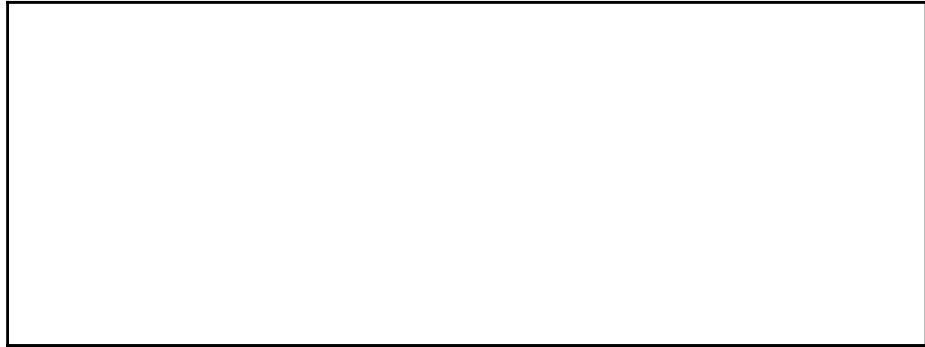
You would like to send a text message (SMS).

Questions for Task 2

If you feel that you cannot respond to a particular item, mark the center point of the scale

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system.	1	2	3	4	5

What did you particularly like about this menu?



What did you dislike about this menu?



Task 3

You want to listen to the radio.

Questions for Task 3

If you feel that you cannot respond to a particular item, mark the center point of the scale

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system.	1	2	3	4	5

What did you particularly like about this menu?

What did you dislike about this menu?

Task 4

You have missed several calls. Now you want to see all of these missed calls.

Questions for Task 4

If you feel that you cannot respond to a particular item, mark the center point of the scale

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

What did you particularly like about this menu?

What did you dislike about this menu?

Task 5

You want to turn on the bluetooth.

Questions for Task 5

If you feel that you cannot respond to a particular item, mark the center point of the scale

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system.	1	2	3	4	5

What did you particularly like about this menu?

What did you dislike about this menu?

Overall-Questions

Please specify your perception of the different menus:

	Liked it very much	Rather liked it	Neither liked it, nor disliked it	Rather disliked it	Disliked it very much	Don't know
Menu of Task 1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Menu of Task 2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Menu of Task 3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Menu of Task 4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Menu of Task 5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Which menu did you like the most and why?

Which menu did you like least and why?

D.3 User questionnaire

Evaluation on cultural preferences in mobile phone interfaces

User questionnaire

Any information you provide is going to be treated anonymously.

Sex: ☐ female
☐ male

Age: _____ Profession/Education(course): _____

What is the name of the country where you currently live?

How long have you lived in this country?

Years: _____ Months: _____

If you have stayed in another country for more than four weeks, please provide here the names of the countries and how long you have stayed there.

Country 1: _____ Years: _____ Months: _____

Country 2: _____ Years: _____ Months: _____

Country 3: _____ Years: _____ Months: _____

Country 4: _____ Years: _____ Months: _____

Country 5: _____ Years: _____ Months: _____

Country 6: _____ Years: _____ Months: _____

Country 7: _____ Years: _____ Months: _____

**What kind of mobile phone do you currently have? (Nokia, Samsung etc.)
If possible provide the model number? (Nokia 6131, Sony Ericsson K800i)**

_____ Years / Months of use _____ / _____

What kind of mobile phones did you have before? (Also provide the model number, if possible)

_____ Years / Months of use _____ / _____

_____ Years / Months of use _____ / _____

_____ Years / Months of use _____ / _____

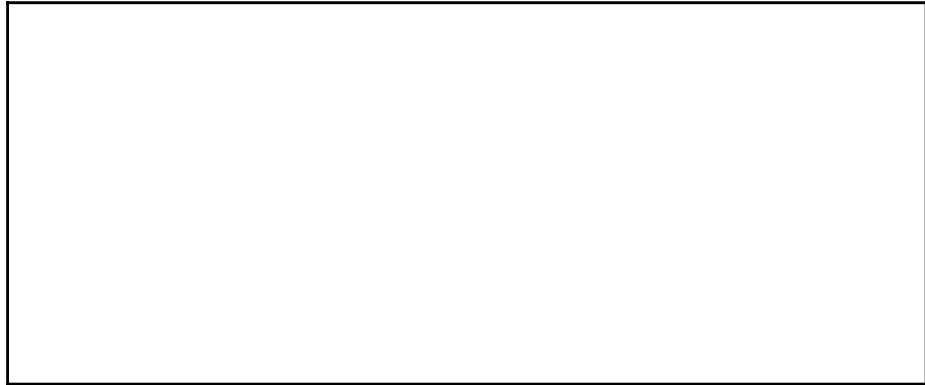
Which functionalities of your mobile phone do you use and how often?

	Daily or less	Three times a week or less	Once a week or less	Once a month or less	Never
Calls	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SMS	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
E-Mail	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Internet	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Games	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Other applications (e.g. calculator or calendar)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Which of the following menu structures have you already used on a mobile phone?

- ☐ List menu
- ☐ Grid menu
- ☐ Tab menu
- ☐ Search menu
- ☐ Tree menu

If you would like to share any other comments about the application or the evaluation, you can provide them here:

A large, empty rectangular box with a thin black border, intended for users to provide additional comments or feedback.

Thank you very much for your participation!

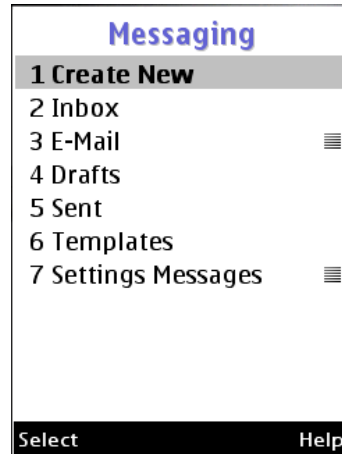
E

Appendix: Screenshots of the menus used for the evaluation

One of the following screenshots was shown to the participants of the evaluation after each task, so they could better remember what the menu looked like. More details about the evaluation sequence can be found in section 6.2.2.



(a) SearchMenu



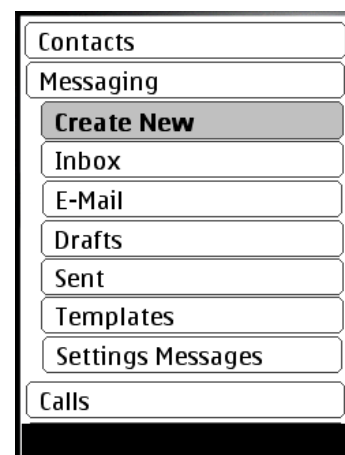
(b) ListMenu



(c) GridMenu



(d) TabMenu



(e) TreeMenu

Figure E.1: Screenshots of the composed menus for the evaluation

F

Appendix: Installation notes

The binaries on the enclosed CD called Campa.jar and Campa.jad need to be copied to the cell phone. The application is started by selecting the jar or the jad file. This depends on the cell phone used. It is recommended to use a Sony Ericsson K800i because the binary was compiled for and tested on that cell phone type. You should then see three midlets ready for selection:

- **01CAMPA:** This is the adaptive application, which presents a menu navigation based on the cultural background of the user and also saves his preferences on the cell phone. A dummy PersonPreference was included in the domain model. Without that PersonPreference the application would directly start to create a new profile and ask the user for his name.
- **02PreferenceSelectionMidlet:** This midlet gives the possibility to compose a mobile phone menu based on the different menu types, the user interface components and the hierarchies. In the first screen the user selects which components he would like to see in the menu types. After proceeding the composed menu is shown.
- **03EvaluationSelectionMidlet:** This midlet was used for the evaluation. One of the five composed menus for the evaluation can be selected.

G

Appendix: Enclosed CD

The enclosed CD has the following content:

- The PDF version of this thesis.
- **CAMPA** contains three folders.
 - **binaries** contains the jar and jad files of the compiled application to install on a cell phone.
 - **nanoxml-j2me** contains the used XML parser as source code and compiled library as jar file.
 - **netbeans** contains the whole netbeans project folder.
- **Documents** contains the OpenDocument Spreadsheet **table_ui_composition.ods** for fast calculation of the Hofstede scores for composed menus and to compare them with Hofstede scores of countries. Moreover the **questionnaires** used for the evaluation are provided and the **results** from the evaluation and their analysis.
- **Domain_model** contains the domain model ontology as an OWL file without any dummy PersonPreferences. A domain model ontology with a dummy PersonPreference individual is provided in the netbeans folder.
- **J2ME_Polish** contains the installation file for J2ME Polish, which has to be installed first in order to compile the source code.

List of Figures

1.1	Charts showing the change in mobile phone subscribers and personal computers	2
1.2	Chart showing the number of mobile phone subscribers per personal computer	2
2.1	Screenshots of Sony Ericsson K800i	6
2.2	Screenshots of Sanyo A5514SA	7
2.3	Screenshots of Anycall SCH-V840	8
4.1	The domain model ontology	22
5.1	Program flow of CAMPA	26
5.2	The process of the profile selection	27
5.3	The process of the questionnaire, which detects current and former countries the user has lived in . .	28
5.4	All user interface components in two ListMenus	30
5.5	The available breadcrumb trails	31
6.1	Different scores in the Hofstede dimensions for Switzerland and China	35
C.1	All screenshots of Sony Ericsson K800i	61
C.2	All screenshots of Nokia 6131	63
C.3	Screenshots of DoCoMo FOMA N905i	65
E.1	Screenshots of the composed menus for the evaluation	86

List of Tables

3.1	Concrete guidelines for designing multi-cultural mobile user interfaces	15
3.2	Rough overview of assignment of components and Hofstede values	17
5.1	Pseudo-algorithm to calculate the Hofstede Values of a user	27
5.2	Possible assignments of components to menus	29
5.3	Change of Hofstede scores depending on used components	32
5.4	Pseudo-algorithm for calculating the Hofstede values of a combined menu	32
6.1	Overview of the results of the pretest	38
6.2	Overview of the results of the maintest - Chinese group	40
6.3	Overview of the results of the maintest - Swiss group	42

Bibliography

- [Badre, 2000] Badre, A. (2000). The Effects of Cross Cultural Interface Design Orientation on World Wide Web User Performance. *GVU Tech Report*.
- [Barber and Badre, 1998] Barber, W. and Badre, A. (1998). Culturability: The Merging of Culture and Usability. *Human Factors and the Web*.
- [Brooke, 1996] Brooke, J. (1996). SUS: a 'quick and dirty' usability scale. *Usability Evaluation in Industry*.
- [Burgmann et al., 2006] Burgmann, I., Kitchen, P., and Williams, R. (2006). Does culture matter on the web? *Marketing Intelligence & Planning*, 24(1):62–73.
- [Cha et al., 2005] Cha, H., Oshlyansky, L., and Cairns, P. (2005). Mobile phone preferences and values: The u.k. vs. korea. *IWIPS '05: Int. Workshop on Internationalisation of Products & Systems*.
- [Chan and Bergen, 2005] Chan, T. and Bergen, B. (2005). Writing Direction Influences Spatial Cognition. *Proceedings of the Twenty-Seventh Annual Conference of the Cognitive Science Society*.
- [Choi et al., 2005] Choi, B., Lee, I., Kim, J., and Jeon, Y. (2005). A qualitative cross-national study of cultural influences on mobile data service design. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 661–670, New York, NY, USA. ACM.
- [Chung et al., 2000] Chung, L., Shih, H., and Goonetilleke, R. (2000). Effect of cultural background when searching Chinese menus. *Proceedings of APCHI/ASEAN Ergonomics*, pages 237–243.
- [Dong and Salvendy, 1999] Dong, J. and Salvendy, G. (1999). Designing menus for the Chinese population: horizontal or vertical? *Behaviour & Information Technology*, 18(6):467–471.
- [Donner, 2007] Donner, J. (2007). The rules of beeping: Exchanging messages via intentional "missed calls" on mobile phones. *Journal of Computer-Mediated Communication*, 13(1):1–22.
- [Dreyfuss, 1984] Dreyfuss, H. (1984). *Symbol Sourcebook: An Authoritative Guide to International Graphic Symbols*. John Wiley and Sons.
- [Fernandes, 1995] Fernandes, T. (1995). *Global interface design: a guide to designing international user interfaces*. Academic Press Professional, Inc., San Diego, CA, USA.
- [Findlater and McGrenere, 2004] Findlater, L. and McGrenere, J. (2004). A comparison of static, adaptive, and adaptable menus. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 89–96, New York, NY, USA. ACM.
- [Ford and Gelderblom, 2003] Ford, G. and Gelderblom, H. (2003). The effects of culture on performance achieved through the use of human computer interaction. pages 218–230.

- [Goonetilleke et al., 2002] Goonetilleke, R., Lau, W., and Shih, H. (2002). Visual search strategies and eye movements when searching Chinese character screens. *International Journal of Human-Computer Studies*, 57(6):447–468.
- [Goren-Bar and Kuflik, 2004] Goren-Bar, D. and Kuflik, T. (2004). Don't miss-r —: recommending restaurants through an adaptive mobile system. In *IUI '04: Proceedings of the 9th international conference on Intelligent user interfaces*, pages 250–252, New York, NY, USA. ACM.
- [Gould et al., 2000] Gould, E. W., Zakaria, N., and Yusof, S. A. M. (2000). Applying culture to website design: a comparison of malaysian and us websites. pages 161–171.
- [Gu et al., 2007] Gu, T., Kwok, Z., Koh, K. K., and Pung, H. K. (2007). A mobile framework supporting ontology processing and reasoning. http://www1.i2r.a-star.edu.sg/~tgu/gutao/paper/RSPSI'07_gutao.pdf, 2008-01-25.
- [Hage, 2007] Hage, M. (2007). System-Befehle sowie SMS + email + Telefon-Funktion in andere Programme einbinden - howto für PDA mit Windows Mobile (Teil 2). <http://pdasammelsurium.blogspot.com/2007/10/system-befehle-sowie-sms-email-telefon.html>, 2007-11-27.
- [Heimgärtner, 2007a] Heimgärtner, R. (2007a). A tool for cross-cultural human computer interaction analysis. *Usability and Internationalization. HCI and Culture*, 4559:8998.
- [Heimgärtner, 2007b] Heimgärtner, R. (2007b). Towards cultural adaptability in driver information and -assistance systems. *Usability and Internationalization. Global and Local User Interfaces*, 4560:372–381.
- [Herman, 1996] Herman, L. (1996). Towards Effective Usability Evaluation in Asia: Cross-Cultural Differences. In *OZCHI '96: Proceedings of the 6th Australian Conference on Computer-Human Interaction (OZCHI '96)*, page 135, Washington, DC, USA. IEEE Computer Society.
- [Hofstede, 2001] Hofstede, G. (2001). *Culture's Consequences: Comparing Values, Behaviors, Institutions, and Organizations Across Nations*. Sage Publications.
- [Hofstede, nd] Hofstede, G. (n.d.). United States - US or American Geert Hofstede Cultural Dimensions Explained. http://www.geert-hofstede.com/hofstede_united_states.shtml, 2008-01-16.
- [ITU, 2006] ITU (2006). International Telecommunication Union - World Telecommunication Development Report and database.
- [Jena, nd] Jena (n.d.). Jena - a semantic web framework for java. <http://jena.sourceforge.net/>, 2008-01-21.
- [Jones and Marsden, 2006] Jones, M. and Marsden, G. (2006). *Mobile interaction design*. John Wiley & Sons. 0-470-09089-8.
- [Kamentz, 2006] Kamentz, E. (2006). *Adaptivität von hypermedialen Lernsystemen*. Phd thesis, University of Hildesheim.
- [Kim and Lee, 2005] Kim, J. H. and Lee, K. P. (2005). Cultural difference and mobile phone interface design: icon recognition according to level of abstraction. In *MobileHCI '05: Proceedings of the 7th international conference on Human computer interaction with mobile devices & services*, pages 307–310, New York, NY, USA. ACM.
- [Lee and Lee, 2007] Lee, S. E. and Lee, G. (2007). K-menu: a keyword-based dynamic menu interface for small computers. In *CHI '07: CHI '07 extended abstracts on Human factors in computing systems*, pages 2543–2548, New York, NY, USA. ACM.
- [LISA, 2007] LISA (2007). The Localization Industry Standards Association - Globalization Industry Primer.

- [Lonely Cat Games, nd] Lonely Cat Games (n.d.). X-plore. <http://www.lonelycatgames.com/?app=xplore>, 2008-02-12.
- [MacKenzie, 2008] MacKenzie, S. (2008). Research note: Within-subjects vs. between-subjects designs: Which to use? <http://www.yorku.ca/mack/index.html>, 2008-04-08.
- [Marcus, 2001] Marcus, A. (2001). Cross-cultural user-interface design. In *Proceedings, Vol. 2, Human-Computer Interface Internat. (HCI) Conf., 5-10 Aug., 2001, New Orleans, LA, USA*, pages 502–505. Lawrence Erlbaum Associates.
- [Marcus, 2002] Marcus, A. (2002). Mapping user-interface design to cultural dimensions. <http://enterpriseinnovator.com/index.php?articleID=985§ionID=13>, 2008-03-06.
- [Marcus, 2005] Marcus, A. (2005). User Interface Design and Culture. *Usability and Internationalization of Information Technology*, pages 51–78.
- [Marcus, 2006] Marcus, A. (2006). Cross-cultural user-experience design. *Diagrammatic Representation and Inference*, 4045:16–24.
- [Marcus et al., 1999] Marcus, A., Armitage, J., Volker, F., and Guttman, E. (1999). Globalization of User-Interface Design for the Web. *Proceedings of the 5th Conference on Human Factors & the Web*. <http://zing.ncsl.nist.gov/hfweb/proceedings/marcus/>, 2008-01-23.
- [Marcus et al., 2003] Marcus, A., Baumgartner, V.-J., and Chen, E. (2003). User-interface design and culture. In Stephanidis, C., Jacko, J. A., and Harris, D., editors, *Human-computer Interaction: Theory and Practice. Proceedings of the HCI International 2003*, pages 153–157. Lawrence Erlbaum Associates.
- [Marcus and Gould, 2001] Marcus, A. and Gould, E. (2001). Cultural dimensions and global web design: What? so what? now what? In *Proc. 6th Conference on Human Factors and the Web*.
- [Marsella and Yamada, 2000] Marsella, A. and Yamada, A. (2000). Culture and mental health: An introduction and overview of foundations, concepts, and issues. In *Handbook of Multicultural Mental Health : Assessment and Treatment of Diverse Populations*. Academic Press.
- [Masuda and Nisbett, 2006] Masuda, T. and Nisbett, R. (2006). Culture and change blindness. *Cognitive Science*, 30(2):381–399.
- [Microsoft Corporation, nd] Microsoft Corporation (n.d.). Windows mobile features (managed) - msdn library. <http://msdn2.microsoft.com/en-us/library/bb158491.aspx>, 2007-11-27.
- [Nisbett, 2003] Nisbett, R. E. (2003). *The Geography of Thought: How Asians and Westerners Think Differently... and Why*. Nicholas Brealey Publishing.
- [Nokia, nd] Nokia (n.d.). Device details – nokia 6131. <http://www.forum.nokia.com/devices/6131>, 2008-02-15.
- [Nokia Corporation, 2007] Nokia Corporation (2007). S60 platform: Utilizing external application views, version 2.1. http://www.forum.nokia.com/info/sw.nokia.com/id/9c5f21fd-2162-42da-aaf6-17a59b943475/S60_Platform_Application_Views_v2_0_en.zip.html, 2007-11-27.
- [nosleepsoftware, nd] nosleepsoftware (n.d.). Filez file browser. <http://nosleepsoftware.sourceforge.net/index.php?page=filez>, 2008-02-12.
- [NTT DoCoMo, 2007] NTT DoCoMo (2007). N905i user's manual. <http://www.nttdocomo.co.jp/english/support/manual/foma/905i/n905i/index.html>, 2008-03-06.

- [Park et al., 2007] Park, J., Han, S., Park, Y., and Cho, Y. (2007). Usability of Adaptable and Adaptive Menus. *Usability and Internationalization. HCI and Culture*, 4559:405.
- [PPC Utils, nd] PPC Utils (n.d.). Handy tree notes. <http://www.ppcutils.com/products/proddet.php?prodid=1>, 2008-02-12.
- [Protégé-OWL API, nd] Protégé-OWL API (n.d.). <http://protege.stanford.edu/plugins/owl/api/>, 2008-01-21.
- [Reinecke, 2005] Reinecke, K. (2005). Computer Aided Vulgarisation of Development Information in Rwanda Part II: Conceptual Design and Development of an XML-based Ergonomic Multimedia Information and Learning Platform for the Education of Agricultural Advisers in Rwanda. Master's thesis, University of Koblenz, Germany.
- [Reinecke, nd] Reinecke, K. (n.d.). Specifying adaptations for a cultural domain model. Unpublished working paper.
- [Reinecke and Bernstein, 2007] Reinecke, K. and Bernstein, A. (2007). Culturally Adaptive Software: Moving Beyond Internationalization. *Usability and Internationalization. Global and Local User Interfaces*, 4560:201–210.
- [Reinecke et al., 2007] Reinecke, K., Reif, G., and Bernstein, A. (2007). Cultural User Modeling With CUMO: An Approach to Overcome the Personalization Bootstrapping Problem. In *Workshop 9 - Cultural Heritage on the Semantic Web, Proceedings of the 6th International Semantic Web Conference*.
- [Resco, nd] Resco (n.d.). Resco explorer 2007 for smartphone. <http://www.resco.net/smartphone/explorer/default.asp>, 2007-02-12.
- [Röse, 2005a] Röse, K. (2005a). Aspekte der interkulturellen systemgestaltung. In *Mensch und Computer 2005: Kunst und Wissenschaft - Grenzüberschreitungen der interaktiven Art*. München: Oldenbourg Verlag.
- [Röse, 2005b] Röse, K. (2005b). Intercultural Human-Machine Systems: Empirical Study of User Requirements in Mainland China. *Usability and Internationalization of Information Technology*.
- [Shenzhen Keenmas Trading Co., nd] Shenzhen Keenmas Trading Co., L. (n.d.). CDMA Samsung SCH-V840 Manufacturer exporting direct from China. http://keenmas.en.alibaba.com/product/50203602/201017125/Mobile_Phones/CDMA_Samsung_SCH_V840.html - 2008-05-03.
- [Shih and Goonetilleke, 1998] Shih, H. and Goonetilleke, R. (1998). Effectiveness of Menu Orientation in Chinese. *Human Factors*, 40(4):569–577.
- [Sony Ericsson, nd] Sony Ericsson (n.d.). Sony ericsson - mobile phones - specifications - k800i. <http://www.sonyericsson.com/cws/products/mobilephones/specifications/k800i>, 2008-02-15.
- [Studio 7.5, 2006] Studio 7.5 (2006). *Designing for Small Screens*. AVA Publishing.
- [The OWL API, nd] The OWL API (n.d.). <http://owlapi.sourceforge.net/>, 2008-01-21.
- [Van Biljon, 2006] Van Biljon, J. (2006). *A Model for Representing the Motivational and Cultural Factors that Influence Mobile Phone Usage Variety*. PhD thesis, University of South Africa.
- [Walton and Vukovic, 2003] Walton, M. and Vukovic, V. (2003). Cultures, literacy, and the web: dimensions of information "scent". *interactions*, 10(2):64–71.
- [Walton et al., 2002] Walton, M., Vukovic, V., and Marsden, G. (2002). Visual literacy as challenge to the internationalisation of interfaces: a study of South African student web users. *Conference on Human Factors in Computing Systems: CHI'02 extended abstracts on Human factors in computing systems*, 20(25):530–531.

-
- [Zi Corporation, 2008] Zi Corporation (2008). Qix functions. <http://www.zicorp.com/QixFunctions.htm>, 2008-02-25.
- [Ziefle and Bay, 2004] Ziefle, M. and Bay, S. (2004). Mental Models of a Cellular Phone Menu. Comparing Older and Younger Novice Users. *Mobile Human-Computer Interaction-Mobile HCI 2004: 6th International Symposium, MobileHCI 2004, Glasgow, UK, September 13-16, 2004: Proceedings*.