OldGen: Mobile Phone Personalization for Older Adults

Alex Olwal **Dimitris Lachanas** **Ermioni Zacharouli**
School of Computer Science and Communication, KTH (Royal Institute of Technology), Sweden
{alx, lahanas, ermioni}@kth.se

ABSTRACT

Mobile devices are currently difficult to customize for the usability needs of elderly users. The elderly are instead referred to specially designed "senior phones" or software add-ons. These tend to compromise in functionality as they attempt to solve many disabilities in a single solution.

We present OldGen, a prototype framework where a novel concept enables accessibility features on generic mobile devices, by decoupling the software user interface from the phone's physical form factor. This opens up for better customization of the user interface, its functionality and behavior, and makes it possible to adapt it to the specific needs of each individual. OldGen makes the user interface portable, such that it could be moved between different phone hardware, regardless of model and brand. Preliminary observations and evaluations with elderly users indicate that this concept could address individual user interface related accessibility issues on general-purpose devices.

Author Keywords
Accessibility, mobile, user interfaces, elderly, older users,

ACM Classification Keywords
H5.2 [Information interfaces and presentation]: User Interfaces - Graphical user interfaces.

General Terms
Design, Experimentation, Human Factors.

INTRODUCTION

The mobile phone is an important technology for older adults as it can improve communication with relatives and friends while also making them more secure and independent [2, 6]. Elderly have often, however, substantially different requirements than younger users, due to ergonomic, cognitive and usability needs. Mobile phones are becoming overly complex for many users, with increasingly more functionality squeezed into shrinking form factors. Vision impairment, poor hearing, motor difficulties and reduced perception, are common age-related problems that, in combination with usability and hardware miniaturization, make mobile phone usage difficult for the aging population [3, 4, 7]. Device shape and size are often not adapted to people with a weak grip. Buttons can be small, poorly separated and without sufficient tactile feedback for fingers with reduced sensitivity. Many features in modern mobile phones can also be irrelevant or confusing for the elderly. Directly accessible functions, menus with shallow navigation depth and clear feedback may instead be preferred. Difficulties in adjusting to new technologies create a strong preference to purchase mobile devices of the same brand, to avoid having to learn a new user interface (UI) [5].

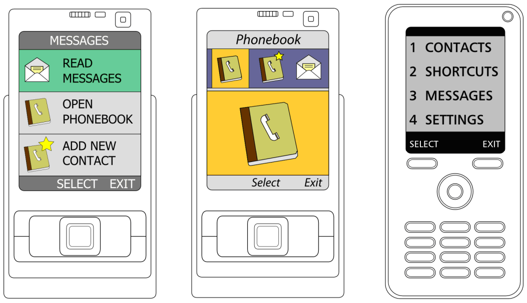


Figure 1. OldGen enables accessible and customizable interfaces on existing mobile phone hardware, by separating user interface and layout from the underlying platform.

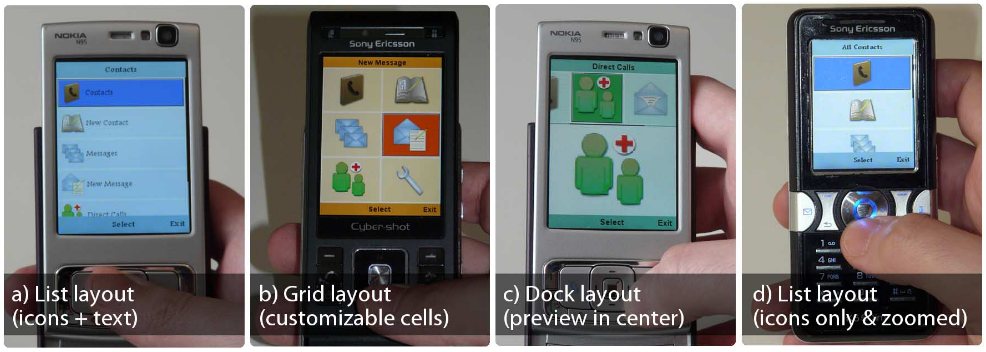
To address these issues, we investigated prioritized functionality for older adults, and a prototype system was developed iteratively with users. OldGen's goal is to allow creating mobile UIs that could be individually customized but still run on general-purpose mobile hardware, as illustrated in Figure 1.

RELATED WORK

Solutions to improve usability for the elderly are, in general, based on dedicated, special-purpose hardware, or on software that is installed on the mobile device to modify its behavior. Many companies market models specifically targeted towards elderly [7], e.g., devices that emphasize ergonomic shape and grip, buttons, big displays and fonts, and dedicated emergency features. Personalization and UI customization, beyond the most basic settings (e.g., ring tones) is uncommon, and the phone and its integrated software are typically intended to be used as preconfigured.

Software-centric approaches focus on vision (e.g., text enlargement or text-to-speech), autonomy (e.g., navigation aids), and personalization and typically depend on the mo-

CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.



a) List layout (icons + text)

b) Grid layout (customizable cells)

c) Dock layout (preview in center)

d) List layout (icons only & zoomed)

Figure 2. Four prototype user interfaces and layouts, specified and generated using the OldGen framework.

bile phone platform that they are installed on. Issues relating to UI layout or complicated menu hierarchies are, however, rarely addressed, as these plugins are meant to add modalities rather than to modify UI or application logic.

Several projects explore the design of mobile UIs. Renaud and van Biljon [10] investigate the needs of elderly and propose an interface design, while Plos and Buisine's case study lead to four physical mockups [9]. MobiDev transforms photographs of hand-drawn UI sketches to Java ME forms, but currently without application logic. De Sá and Carriço provide rapid prototyping directly on Windows and PalmOS devices [1], while Verstockt et al.'s software layer simplifies access to functionality on Windows Mobile [1].

Rather than only supporting smartphones, which are currently less common among our target users, our framework is specifically designed to work on mid- and low-end devices. In our work, we focus on a customizable and extensible layer for the most important phone features using XML specification and platform-independent Java ME.

OLDGEN FRAMEWORK CONCEPT

Our OldGen framework enables the exploration of customizable mobile interfaces for elderly users. The idea is that a family member or caretaker could use a tool (on the desktop or on the device, like [1] or [8]) to design and tweak UI behavior, or look-and-feel based on a library of preconfigured templates. These may include available phone features, menu types, depth in menu hierarchies, font sizes, colors, contrast, brightness and other aspects of the UI that are specific to a user's needs.

PROTOTYPE UIS

Five prototype UIs illustrate how various layouts can be supported in the system's different views, menus and forms. These implementations vary in their combinations of GUI components, customizable appearance and specified behavior. We provide direct access to some functions, and use deeper navigation depth in other. (See Figure 1 and 2.)

SYSTEM ARCHITECTURE

OldGen transforms UI specifications into Java ME applications. Cross-compilation frameworks, such as MoSync (mosync.com), J2ME Polish (j2mepolish.org), and Grapple (grapplemobile.com), are popularly used to provide interoperability for developers. They generate applications for multiple platforms from source code written in a single language. While many use source code as input, OldGen uses XML for UI specifications.

The design is used to generate an XML specification of the UI, which is parsed and converted into source code that can be compiled into a Java ME application. (The XML specifications are currently manually defined, since the visual tool is under development.) Our approach focuses on enabling the UI construction solely through specification, where the basic functionality is available as building blocks in XML.

XML specification layer. XML elements specify application settings and components to ensure flexibility and interoperability as it allows the system to be extensible for new functionality. It also inherently allows future parsers to produce source code for other platforms.

XML-Java Parser. The OldGen framework parses the UI specification (XML) and generates the corresponding Java files. It uses core Java ME, the cross-platform UI components in the Lightweight User Interface Toolkit (LWUIT, http://lwuit.dev.java.net), and a resource file that specifies font and color information.

IMPLEMENTED FUNCTIONALITY

Our intention was to initially support the most prioritized functionality for elderly [2, 4, 5, 7, 9, 10], rather than all possible features:

Calling. The most commonly used function. As in the whole framework, the aim is to both provide easy access and support alternative functionality. The call can be made with physical buttons (e.g., call button) or through the UI (e.g., soft button/GUI button).

Sending/Receiving SMS. Similarly to calling, numbers are directly entered, or selected from the phone book. The send action can be mapped to a soft button or a GUI button.

Phone Book. Stores contacts with common information, such as, name, number, address, relationship, and nick name. Contacts can also be assigned to a list of shortcuts.

Image Storage. Images can be stored in the application, and added to a contact. Photographs may, for example, help elderly easier recall or recognize contacts.

Zoom / Scaling. A scalable UI was implemented as it could address problems with poor vision. Both fonts and images can be scaled to improve legibility. Zoom in/out can currently be mapped to physical keys (e.g., */#, or volume +/–), GUI buttons in the UI, or an option in the global settings menu. (See Figure 3, left.)

GRAPHICAL USER INTERFACE

Five component types and three different layouts are currently available. The UIs have at least one form, where each form has a unique name and includes various components and functions, arranged according in the specified layout.

Layouts. Three layouts can be used to arrange components. The GUI elements can be shown in a list (*List Layout*, Figure 2a and 2d), divided into a grid, (*Grid Layout*, Figure 2b) or accessed in a horizontal menu at the top of the screen with an enlarged preview of the active option in the center (*Dock layout*, Figure 2c).

GUI Components. Our five basic components form the structure of the UI and can be found in most UI frameworks (See Figure 3, right). The *Label* is used for labeling text input areas or images with a text string or image. The *Soft button* activates a function (e.g., "delete contact") when the corresponding physical button underneath the on-screen label is pressed. The *GUI button* is an iconic button that the user navigates to and then activates (not directly accessible with dedicated physical button). The *List* displays items (e.g., contacts or text messages), in a list view. The *Text Area* supports numeric or text input (e.g., phone numbers).

PRELIMINARY EVALUATION

We were interested in evaluating the potential of our customizable UI framework for existing phone hardware. Our evaluations were, however, not meant directly to evaluate the usability of the sample UIs, as these had not been individually customized for participants in the tests.

User observation sessions

Four user observation sessions were conducted in an iterative cycle with interleaved development and redesign of functionality and UIs. The first three sessions were with non end-users to identify technical deficiencies and missing capabilities. Further insights were provided through five individual and informal test sessions with retired seniors (63–74 years old). The combined feedback from these sessions indicated appreciation for scalable UIs, simplicity, and the possibility for customization.

Preliminary user evaluation

Six women, 52–76 years old (avg 68.5, stdev 9), were recruited from a charity organization where retired women meet weekly to sew. The test consisted of a pre-test questionnaire, tasks to be completed in three different mobile UIs, and a post-test questionnaire with feedback and ranking of the UIs. Due to the small number of participants, statistical analysis of the data was not performed.

The participants had different professional backgrounds (children caretaker, post office manager, nurse and teacher, cartographer, engineer, and pedagogue) and had owned 1–4 mobile phones (avg 2.5), which they had used for 1–15 years (avg 6.7). Most participants rarely called from their mobile (4: rarely, 1: sometimes, 1: daily) and only one participant used SMS (only occasionally). As in previous studies [6], all considered calling the most important function.

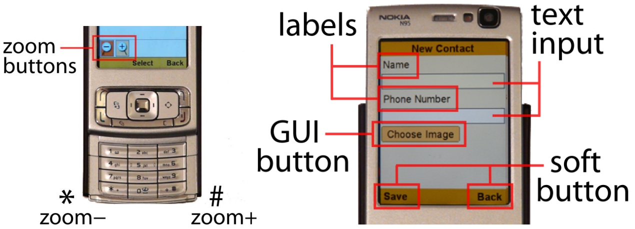


Figure 3. Left: Functionality, e.g., zoom, can be mapped to physical keys, GUI buttons, or soft buttons (not shown). Right: The five currently available GUI components.

Participants used the default UI on a generic phone (Nokia N95), a simplified OldGen-built UI installed on the generic phone, and the default UI on a senior phone (Doro PhoneEasy 410gsm). For each interface, functionality and operation was explained. Participants then performed some basic tasks: adding a contact, writing an SMS, calling the contact, and erasing the contact. The UI interactions were recorded on video, and notes were taken by the study leader.

Results

Generic phone (Overall mean score 2.5)

Participants did not understand the GUI icons, had problems reading text (small font) and complained about insufficient contrast. Physical buttons were poorly separated, flat, slippery, and the text on them was too small. Most participants also found it difficult to understand how to navigate in the UI with the cursor keys. All participants needed assistance to complete the tasks.

Generic phone + OldGen-built UI (Overall mean score 3.4)

Since our custom UI ran on the same hardware, it suffered from the same button issues. Participants were frustrated by the three physical buttons (menu, media, hang up) that exit the OldGen UI, a limitation of Java ME. While zoom was appreciated, it was not always clear how to activate soft buttons, and participants had problems with forms that required scrolling. Several participants did, however, complete the tasks without requiring assistance.

Senior phone (Overall mean score 4.2)

All participants liked the large, separated rubber buttons, the screen's contrast and large font size. Participants did not mind the lack of icons in the UI and appreciated visual feedback for buttons they were pressing while writing. Most participants completed the tasks without assistance. (The SMS tasks were excluded, as the senior phone did not support them.)

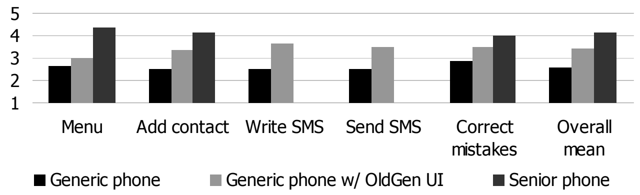


Figure 4. Post-test questionnaire mean values. Participants ranked aspects in the UIs on a 5-point Likert scale (1=very hard, 5=very easy). Senior phone did not support SMS.

Discussion

After analyzing results, we identified numerous changes to the OldGen UI that should improve usability: better integrated zoom feature, increased contrast, no icons, some renamed menu elements, avoiding scrolling, and visual feedback for pressed buttons. All of which can be changed in software. A limitation that cannot be addressed is the behavior of the three buttons that exit the OldGen UI.

While the OldGen UI performed worse than the senior phone, it was more appreciated than the original N95 UI (See Figure 4). We believe that an improved OldGen UI, installed on a generic phone with better keypad, should be able to provide a similar level of usability as the senior phone.

The preliminary evaluation is limited by the small number of all-female participants that were recruited from the same context, but seems to indicate the potential for our continued work. We are currently refining the prototypes for an improved version, which will be used in our next experiments and formal evaluations that should also involve long-term studies that could provide better ecological validity.

CONCLUSIONS AND FUTURE WORK

We have introduced the concept of customizable mobile UIs, through our OldGen framework, which separates UI specification and functionality from the underlying hardware. Several prototype UIs were generated in Java ME as a case study, and user interaction with various software generations was observed. The implemented prototype framework seems promising in its ability to help addressing usability problems in today's mobile phones by:

- Introducing flexible customization of mobile UIs that could be tailored to specific individual needs.
- Decoupling software UI from the phone's physical form, allowing users to only having to learn one mobile UI, since a portable UI could be moved between phone hardware, regardless of model and brand.
- Through customization, introducing accessibility features on generic devices, such that elderly would not only be limited to specific "senior phones".

The prototype framework makes it possible to generate a range of different UIs through its specification mechanism. As such, it may be an interesting complement or alternative to hardware-only or other software-based approaches. In this work, we prioritized a set of core features, which can be extended through the extensible architecture. Future work also includes support for platforms that allow more intimate integration with the underlying system, such as Android (android.com), iOS (apple.com/ios), MeeGo (meego.com) and Symbian (symbian.com).

Finally, we would like to emphasize that an advanced implementation of this concept could be beneficial for every mobile user, in the "Design-for-all" spirit. It would make it possible to have portable personal mobile UIs that would evolve through their use on different devices, in different contexts and through gradual experimentation and refinement, towards the optimal behavior for each user.

ACKNOWLEDGEMENTS

We thank Arash Saghafian, Per Sunesson and Magdalena Sjöstrand for running many of our studies. The work was made possible by funds from the Swedish Institute for Assistive Technology and support from SO4IT and Rule Communication, and mobile devices from Doro, Ericsson and Nokia. We also want to thank the National Pensioners' Organization, Red Cross Täby's Design and Activity Group, all participants in our studies, and our reviewers for their valuable feedback, which helped us improve the paper.

REFERENCES

1. De Sá and Carriço. 2009. A mobile tool for in-situ prototyping. *Proc. MobileHCI '09*, Article 20.
2. Fisk A., Rogers W., Charness N., Czaja S. and Sharit J. 2004. Designing for Older Adults: Principles and Creative Human Approaches. *CRC Press ISBN 0-41-528611-5*.
3. Hassan, H. and Nasir, M., H., N. 2008. The use of Mobile Phones by Older adults: A Malaysian study, *Proc. SIGACCESS 2008*, 11–26.
4. Holzinger, A., Searle, G., Nischelwitzer, A. 2007. On Some Aspects of Improving Mobile Applications for the Elderly. *Proc. Universal Access in HCI, HCII 2007*, 923–932.
5. Kim, H., Heo, J., Shim, J., Kim, M., Park, S. and Park, S. 2007. Contextual Research on Elderly User's Needs for Developing Universal Design Mobile Phone. *Proc. Univeral Access in HCI, HCII 2007*, 950–959.
6. Massimi, M., Baecker, R. M., Wu, M. 2007. Using participatory activities with seniors to critique, build and evaluate mobile phones. *Proc. SIGACCESS 2007*, 155–162.
7. Pattison, M., Stedmon, A. 2006. Inclusive design and human factors: designing mobile phones for older users. *PsychNology Journal 4(3)*, 2006, 267–284.
8. Pfleging, B., del Carmen Valderrama Bahamondez, E., Schmidt, A., Hermes, M. and Nolte, J. 2010. MobiDev: a mobile development kit for combined paper-based and in-situ programming on the mobile phone. *CHI '10 Extended Abstracts*, 3733-3738.
9. Plos, O. and Buisine, S. 2006. Universal design for mobile phones: a case study. *CHI '06 Extended Abstracts*, 1229-1234.
10. Renaud, K. and van Biljon, J. 2010. Worth-centred mobile phone design for older users. *Universal Access Information Society, 9, 4*, 387-403.
11. Verstockt, S., Decoo, D., Van Nieuwenhuyse, D., De Pauw, F. and Van de Walle, R. 2009. Assistive smartphone for people with special needs: the personal social assistant. *Proc. HSI'09*, 328-334.